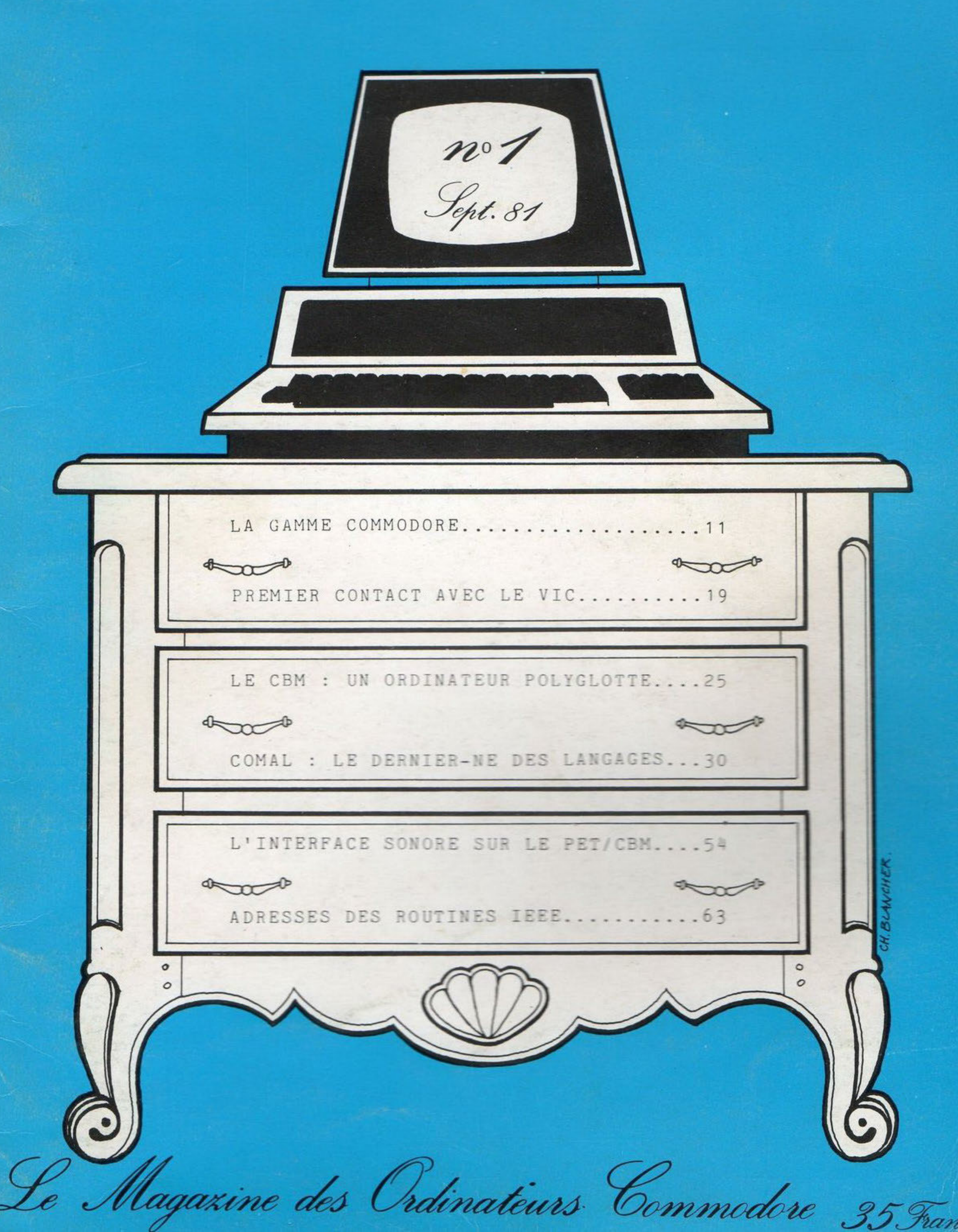
La Commode



Le Magazine des Ordinateurs Commodore

35 Francs



SIDEG INFORMATIQUE

170, rue Saint-Charles, 75015 PARIS. Tél. : 557.79.12
Ouvert du lundi au samedi de 9 h 30 à 18 h 30 sans interruption
Vente sur place et par correspondance — Commande par téléphone — Crédit — Leasing —
Détaxe à l'exportation — Carte Bleue ou Visa

Déménagement

A compter du 8 septembre 1981, nous vous accueillerons 170, rue Saint-Charles à PARIS 15e.
A cette occasion, nous vous offrons une remise exceptionnelle de 5 % pour toute commande de matériel : ordinateur ou périphérique informatique, à valoir sur l'achat de logiciels, librairie ou fournitures diverses.
Valable du 8 septembre au 7 novembre 1981.

COMMODORE SÉRIE 8000



CBM 8032-32 K	11 950,00 F H.T.
CBM 8050	11 950,00 F H.T.
CBM 8026 imprimante à Marguerite avec clavier azerty	11 950,00 F H.T.
TOTAL	35 850,00 F H.T.

SÉRIE 4000



CBM 4016-16 K	7 995 F T.T.C.
CBM 4032-32 K	10 500 F T.T.C.
CBM 4040-360 K	10 500 F T.T.C.
CBM 4022 imprimante	5 900 F T.T.C.
Lecteur de K7	640 F T.T.C.
Imprimante GP 80 M avec	3 500 F T.T.C.

Logiciels professionnels

OZZ Logiciel de gestion de fichiers	2 950 F H.T.
Traitement de texte	2 450 F H.T.
Comptabilité générale	3 500 F H.T.
Paie	2 450 F H.T.
Visicalc - Logiciel de calcul et d'aide à la décision	950 F H.T.

OZZ est un logiciel d'écriture de programmes d'applications, demandant seulement l'entrée des paramètres de travail.
Le logiciel OZZ permet d'écrire des programmes sur mesure pour informatiser facilement les nombreux problèmes auxquels se trouvent inexorablement confrontées les entreprises, grandes ou petites.
• Analyse financière, • Tableau de bord, • Courriers, mailings, • Catalogues, • Agendas, • Gestion des ventes, • Trésorerie, • Tenue de stock, • Facturation, • etc...

Il ne s'agit là que de quelques exemples. La « flexibilité » et la « puissance créative » du logiciel OZZ permettent pratiquement d'en multiplier à l'infini ses applications pratiques.

TRAITEXT est un logiciel de traitement de textes très perfectionné, qui va simplifier des travaux de secrétariat et vous donner une efficacité supplémentaire.
Fonction stockage :
- Environ 4 pages (11 000 caractères) en mémoire centrale,
- Environ 180 pages (500 000 caractères) sur chaque disquette interchangeable,
- Correction caractère par caractère ou mot par mot,
- Substitution automatique d'un mot à un autre mot,
- Remplacement, suppression ou déplacement de paragraphes,
- Insertion automatique de variables issues d'un fichier,
- Justification, centrage, identification, etc...
- Taille de la page, valeur de l'interlignage et de l'écartement des caractères,
- Soulignage, surimpression et césure optionnelle,
- Arrêts d'impression (pour changement de marguerite par exemple),
- Non-impression d'une partie de texte, fusion de plusieurs textes, pagination automatique, etc...,
- Frappe au clavier d'un texte pendant l'impression d'un autre texte, etc...

Extensions spécialisées pour CBM

Interface haute résolution	3 500 F T.T.C.
Synthétiseur de voix	N.C.
Paddles ou manches à balai	650 F T.T.C.
Interface sonore à 4 voies	650 F T.T.C.
Interface vidéo	500 F T.T.C.
ROM Edex	529 F T.T.C.
ETC...	

Logiciels divers

Tenue de compte - K7	195 F T.T.C.
Gestion de fichier - K7	150 F T.T.C.
Applications pratiques - K7	100 F T.T.C.
Les mouches de l'espace - K7	150 F T.T.C.
Trek-X - K7	100 F T.T.C.
Flipper - K7	70 F T.T.C.
Micro-Chess 2.0 - K7	150 F T.T.C.
Dames Challenger - K7	195 F T.T.C.
Invaders	90 F T.T.C.
ETC...	

Nos prix sont donnés à titre indicatif et peuvent être modifiés sans préavis.

CATALOGUE GRATUIT SUR SIMPLE DEMANDE

$$$$$$$$$$$$
$ $
$ SOMMAIRE $
$ $
$$$$$$$$$$$$



Voici le premier numéro de La Commode, le magazine des ordinateurs Commodore. Il est entièrement consacré aux ordinateurs Commodore, à toute la gamme Commodore, du PET 2001 au CBM 8032 en passant par les 3000 et 4000. Il concerne aussi le VIC, le CBM 8096, le CBM 9000 Micro-mainframe et tout nouveau modèle susceptible d'apparaître, ainsi que les périphériques.

Il s'adresse à tous les utilisateurs, du débutant au plus chevronné. Vous trouverez des trucs, des astuces, des adresses mais aussi des explications de fond. Les instructions seront expliquées une par une afin de former une encyclopédie pour les débutants, mais à laquelle les vétérans pourront toujours se référer.

La Commode est indépendante de tout constructeur. Nous publierons des évaluations et des bancs d'essai des nouveaux modèles, des logiciels, des périphériques non Commodore.

Voici une liste non limitative des différentes rubriques qui apparaîtront régulièrement:

-Nouveautés-Actualités. Tous les potins sur les annonces,les salons...

-Langages. En principe BASIC, Assembleur, PASCAL mais aussi COMAL, FORTH, LISP...

-Hardware-Interfaces. Des montages, des manips. A vos fers à souder!

-Applications. Traitez vos problèmes dans le monde réel. Gérez efficacement votre entreprise.

-Disques et périphériques. Des bancs d'essai, mais aussi des conseils, des astuces d'utilisation et des précautions à prendre.

Nous avons aussi une rubrique enseignement et "les jeux cousus-main de Jacques COUTHURES".

La rubrique bibliographie est représentée dans ce premier numéro par un rappel des articles qui ont été consacrés au PET/CBM dans l'Ordinateur Individuel et dans Microsystèmes.

Mais nous vous rappelons que c'est vous, lecteurs, qui ferez véritablement ce magazine, en nous envoyant vos opinions, vos idées, vos trucs, vos programmes. Ce magazine doit être un lieu d'échange entre utilisateurs Commodore et distributeurs, et non un recueil de cours magistraux faits par la Rédaction. A vos plumes!

Bien sûr, il y aura un courrier des lecteurs. Nous voulons bien aussi publier gratuitement vos annonces - honnêtes!- mais attention au délai entre les numéros! Enfin nos colonnes sont ouvertes aux clubs et à tous ceux qui souhaitent former un club. Contactez-nous.

Daniel-Jean DAVID

COMMODORE LEADER EURO



C'est par l'innovation continue, fruit de la maîtrise des techniques de pointe, que Commodore a conquis sa place de leader européen de la micro-informatique. Cette position, bien connue de tous ceux qui suivent de près l'actualité informatique, vient d'être corroborée par la publication d'une récente étude de marché conduite par la firme indépendante I.D.C. (1).

La conséquence la plus notable de cette brillante performance, s'est traduite par un renforcement des structures européennes de Commodore aux plans marketing et fabrication.

C'est ainsi que Commodore a récemment ouvert **une usine en R.F.A., à Braunschweig,** afin de répondre aux besoins croissants du marché européen.

A quelles qualités particulières doit-on attribuer le succès des micro-ordinateurs de la gamme Commodore ?

- **la fiabilité exceptionnelle,** résultat d'une électronique particulièrement bien conçue et fabriquée avec beaucoup de soins à partir de composants de qualité,
- **l'excellent rapport prix performances ;** Commodore vous en donne plus pour votre investissement,
- **la parfaite homogénéité de la gamme** et l'architecture décentralisée des systèmes où chaque périphérique est intelligent,
- **des produits toujours "dans le vent",** qui bénéficient des unités intégrées de recherche, de développement et de production mises en place par Commodore depuis 1975 ; la gamme Commodore répond ainsi aux besoins variés et croissants du marché,
- **les maxi-services Procep,** qui apportent aux utilisateurs français de la gamme Commodore, un environnement adapté à leurs besoins. Une plus-value très importante et très appréciée.

(1) I.D.C. Europa Ltd, 2 Bath Road, LONDON W 41 LN.



La gamme Commodore, harmonieuse et complète, répond aux besoins les plus divers des débutants comme des virtuoses de la micro-informatique.

COMMODORE VIC 20

Entrez dans l'univers fabuleux de la micro-informatique avec le Commodore VIC 20*.

Oui, initiez-vous avec un **vrai** micro-ordinateur, puissant et évolutif... et cependant à un prix époustouflant. Le VIC 20 vous permet de donner libre cours à votre créativité en disposant d'un micro-ordinateur de qualité, bien à vous, chez vous.

* le Commodore VIC 20 est disponible en version PAL ; ultérieurement une version SECAM, développée par Procep, sera mise sur le marché.

SERIE COMMODORE CBM 4000

Professions libérales, Laboratoires, Centres de Recherche, Organismes de Formation, etc., investissez clairvoyant. Voyez le système complet et homogène Commodore CBM 4001.

Avec
- son unité centrale de 32 Ko (CBM 4032),
- son unité de double minidisquette de 360 Ko (CBM 4040),
- son imprimante à traction (CBM 4022),

le système CBM 4001 et ses logiciels standard disponibles, vous offrent de nombreuses possibilités pour résoudre vos problèmes.

NB : Une unité centrale de 16 Ko (CBM 4016) est également disponible.

SERIE COMMODORE CBM 8000

PME/PMI, services décentralisés ou autonomes des Grandes Entreprises, investissez dans le haut de gamme. Voyez le système complet et homogène Commodore CBM 8001.

Vous disposerez d'un équipement qui, associé aux logiciels de haut niveau proposés par Procep, vous apportera une gestion efficace et vous donnera les meilleures chances de rester compétitif.

Dans le cadre de sa politique d'apport de plus-value par la francisation, **Procep équipe gratuitement les CBM 8001 d'une ROM supplémentaire : Edex 4.0,** qui complète le Basic d'origine de l'unité centrale.

Bientôt **2 nouveautés** viendront enrichir la série CBM 8000 :
- une unité centrale (CBM 8096) mettra à votre disposition 96 Ko RAM,
- une nouvelle unité centrale vous donnera accès, en plus des langages actuellement disponibles (Basic, Comal, Pascal, Pilot, Lisp, Forth), à de nouveaux langages tels que APL, Cobol, Fortran, etc.



MASTER

Avec **Master,** les "développeurs" de logiciels vont être comblés.

Master est un ensemble d'utilitaires qui vous permet :
- de réaliser sur le système CBM 8001 des développements et des mises au point de programmes qui ne pouvaient l'être jusqu'à présent que sur des "gros ordinateurs",
- de protéger les logiciels ainsi développés contre d'éventuelles copies.

Ainsi **Master,** non seulement vous facilite la vie, mais protège votre investissement "matière grise".

Master comprend :
- Master file (séquentiel indexé),
- Master screen (générateur d'écran),
- Master plus (extension Basic, multiprécision, compactage et décompactage, etc.),
- Master print (générateur d'édition).

EDEX 2.0

Destinée aux unités centrales de la série CBM 3000, **Edex 2.0** est une ROM supplémentaire venant compléter le Basic d'origine. Commandes et fonctions complémentaires apportées par **Edex 2.0 :** Auto, Renu, Delete, Apend, Dump, Error, Find, Call, If then else, Print using, Plot, Reset, Beep.



PROCEP DISTRIBUTEUR EXCLUSIF POUR LA FRANCE DE COMMODORE

PEEN DE LA MICRO-INFORMATIQUE

SYSTEME CBM 8001

Imprimante à traction CBM 8024
132 colonnes, 160 c/s bidirectionnelle

Unité centrale CBM 8032
32K octets RAM, écran 80 colonnes

Unité de double disquette CBM 8050
capacité 2 x 500 K octets

EDEX 4.0

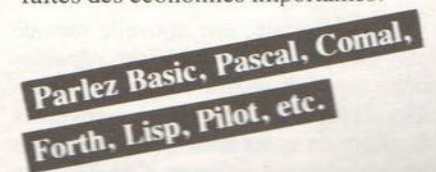
Edex 4.0 est une ROM venant compléter le Basic d'origine de l'unité centrale CBM 8032. Commandes et fonctions complémentaires apportées par Edex 4.0 - particulièrement utiles pour améliorer ou optimiser les logiciels de haut niveau développés sur le système CBM 8001 - : Auto, Renu, Delete, Merge, Dump, Error, Find, Fetch, Call, If then else, Print using, Plot, Reset, Hard copy, Beep.
De plus, un exécuteur pas à pas facilite la mise au point des programmes.

TRIEX

Résidant sur ROM, **Triex** vous offre les possibilités des "grands" systèmes en matière de tri et de gestion de fichiers.

MULTEX

Résidant sur ROM, **Multex** vous permet de connecter un même périphérique à plusieurs unités centrales CBM 8032.
Par exemple, plusieurs CBM 8032 répartis dans différents services peuvent travailler sur un même fichier se trouvant sur la même disquette ou imprimer sur la même imprimante.
Avec **Multex** vous gagnez de la place et vous faites des économies importantes.



Le Basic est le langage le plus employé sur les systèmes Commodore. Mais vous pouvez changer de langage aussi facilement que de disquette.

Pour tous les produits décrits ci-dessus, une documentation technique est disponible chez les Distributeurs-Revendeurs Procep dont la liste figure dans les pages qui suivent.



3 STANDS PROCEP

pour vous accueillir quel que soit le but de votre visite.

Pour les fanas de la micro-informatique comme pour les débutants :
SICOB-BOUTIQUE
(stands n° 124, 126 et 128).

Pour les spécialistes du domaine industriel :
SICOB O.E.M.
(stands n° 157 et 159).

Pour les chefs d'entreprise, les cadres et les gestionnaires :
SICOB PALAIS du CNIT
(niveau 3 - stand 3 C 3308)

►►►►►





PROCEP DISTRIBUTEUR EXCLUSIF POUR LA FRANCE DE COMMODORE



PROCEP. MAXI SER

La "francisation" Procep : pour la satisfaction des utilisateurs

Procep a su investir, dès sa création, dans la "francisation" des micro-ordinateurs Commodore en leur apportant une valeur ajoutée particulièrement importante au niveau des services offerts aux utilisateurs.

- ***information :*** documentation, manuels et bulletin des utilisateurs de la gamme Commodore,
- ***formation :*** séminaires et cours destinés aux utilisateurs, revendeurs et distributeurs ainsi que des sessions spéciales pour l'enseignement,
- ***développement :*** logiciels de base, utilitaires, logiciels d'application de haut niveau pour la gestion et la bureautique, cartes industrielles, interfaces, etc.,
- ***garantie*** portée à un an, au lieu des 3 mois offerts par le constructeur.

Des logiciels à la hauteur

Comptabilité générale

Le logiciel de comptabilité générale fonctionnant sur CBM 8001 a été conçu et longuement testé par une équipe de professionnels expérimentés en comptabilité et en informatique. Simple d'emploi, il permet aux PME/PMI de résoudre efficacement leurs problèmes comptables.

Paie

Outre le calcul et l'établissement des bulletins de paie, le logiciel de paie permet la définition et l'actualisation en mode conversationnel du fichier du personnel et du plan de paie de l'entreprise et des salariés.

Après mise à jour automatique, vous pouvez demander au logiciel l'édition instantanée ou différée des éléments propres à la paie d'une période donnée, ainsi que des informations nécessaires aux documents fiscaux, administratifs et comptables.

Traitement de texte

Tout en conservant la potentialité des capacités de l'ordinateur pour traiter les problèmes de gestion, les systèmes CBM de Commodore apportent avec le **logiciel de haut niveau Traitext,** des solutions "traitement de texte" particulièrement efficaces et à des prix "micro-informatique".

Bien entendu, le logiciel Traitext est entièrement francisé et le clavier standard du CBM 8001 a été modifié de façon à le rendre similaire à celui auquel sont habituées les secrétaires : **le clavier AZERTY avec voyelles accentuées.**



De façon à apporter **la** solution traitement de texte qui correspond à vos besoins, Procep vous propose 3 formules originales :

- configuration **spécialisée** avec imprimante à marguerite,
- configuration **double** pour la gestion et le traitement de texte, avec imprimante à aiguilles et imprimante à marguerite,
- configuration **mixte** avec imprimante à aiguilles.

L'imprimante à marguerite CBM 8027 peut être remplacée par l'imprimante CBM 8026. Elle présente l'avantage d'être équipée d'un clavier et en conséquence, de pouvoir être utilisée comme une machine à écrire de bureau.

Dès à présent, Traitext fait le lien avec le logiciel de calcul Visicalc. Dans un proche avenir, une liaison avec logiciels de gestion de fichiers sera disponible.

Une nouvelle race de logiciels.

Avec Ozz et Visicalc, une nouvelle race de logiciels est née. Des logiciels capables d'apporter aux utilisateurs, même non informaticiens, la faculté d'écrire des programmes d'application, en ayant seulement à entrer les paramètres de travail au cours d'un dialogue homme-machine.

OZZ **: pour écrire des programmes sur mesure** afin d'apporter **votre** solution informatisée à **vos** problèmes de gestion courants. Avec Ozz, il est maintenant possible d'informatiser, sans se créer de complications supplémentaires, de nombreuses applications en définissant seulement les paramètres de travail. Nul besoin d'être informaticien pour mettre en œuvre le logiciel Ozz.

Ozz comporte une gestion automatique de fichiers permettant de réaliser des sélections très élaborées d'enregistrements, quel que soit le nombre de critères choisis.

VISICALC **: un puissant outil de planification et de prévision.**

Visicalc est en quelque sorte un bloc-note électronique dont chaque feuille se présente comme une grille composée de rangées et de colonnes. Chaque case peut être adressée comme une coordonnée : elle contiendra un label ou une valeur. Il vous suffit de modifier un paramètre et tous les résultats sont modifiés en conséquence. C'est l'outil idéal pour établir des budgets, modifier des prévisions, et réaliser toutes sortes d'analyses financières ainsi que des calculs techniques et scientifiques.


COMMODORE LEADER EUROPEEN DE LA MICRO-INFORMATIQUE

VICES POUR LA MICRO

Le bon choix à votre porte.

Procep renforce régulièrement son réseau de distributeurs et de revendeurs auxquels il apporte le maximum d'assistance.

02 SIGMATIQUE
10, rue du 13-Octobre, **02000 LAON,**
T. (23) 20.12.69

06 OLLIVIER, 20 bis, rue Gubernatis,
06000 NICE, T. (93) 85.33.17
COMPUTERLAND
Av. Léon Béranger,
06700 SAINT-LAURENT-DU-VAR,
T. (93) 07.61.12
ONDE MARITIME, 28, bd du Midi,
06150 CANNES, T. (93) 47.44.30

13 EUROPE ELECTRONIQUE
13, bd du Redon "La Cloris",
13001 MARSEILLE, T. (91) 82.07.91
COMPUTERLAND
1, av. de Corinthe,
13006 MARSEILLE,
T. (91) 78.02.02

14 D.P.C. NÉGOCE, 10-16 rue St-Jean,
14000 CAEN, T. (31) 85.40.11

16 P.M.E. INFORMATIQUE
31, rue du Sauvage,
16000 ANGOULEME, T. (45) 38.32.97

20 GRIMALDI, 41, cours Napoléon,
2 A 000 AJACCIO, T. (95) 21.23.65

21 SETTEM, 36, rue Jeannin,
21000 DIJON, T. (80) 66.16.43

24 C.C.S., 7, rue de la Constitution,
24000 PÉRIGUEUX, T. (53) 09.45.16

25 REBOUL, 34, rue d'Arène,
25000 BESANÇON, T. (81) 81.02.19

27 SEM INFORMATIQUE
55-61, rue F.-Roosevelt, BP. 872,
27008 EVREUX, T. (32) 39.26.08

28 ASSIST. INFORM. OUEST
9, rue du Bois-Sabot, **28100 DREUX,**
T. (37) 46.25.42

29 ARMOR BUREAU,
Le Grand Guélen,
Route de Rosporden,
29105 QUIMPER CEDEX,
T. (98) 90.08.29

31 MIDI MICRO INFORM.
26, rue M.-Fonvieille,
31000 TOULOUSE, T. (61) 23.68.50
SOUBIRON, 9, rue J.-F.-Kennedy,
31000 TOULOUSE, T. (61) 21.64.39

33 GEDIF, 14, cours d'Albret,
33000 BORDEAUX, T. (56) 44.50.97
ARTIS TECHNIQUE
10 bis, pl. des Grands-Hommes,
33000 BORDEAUX, T. (56) 81.22.26
AQUITAINE MICRO INF.
134, bd F.-Roosevelt,
33000 BORDEAUX, T. (56) 91.78.74

34 MICRO SCOP, 15, cours Gambetta,
34100 MONTPELLIER,
T. (67) 27.53.09

35 A.B.M., 5, rue Louis-Turban,
35000 RENNES, T. (99) 50.50.42
COMPUTERLAND
13, av. du Mail,
35000 RENNES, T. (99) 54.47.12

37 COGEC, 54, rue du Général Renault,
37000 TOURS, T. (47) 20.72.04

38 I.T.E.C./SI-TEL, 5, av. Victor Hugo,
38130 ECHIROLLES, T. (76) 23.07.27

42 E.T.F./SI-TEL, 77, rue de Marengo,
42029 SAINT-ETIENNE CEDEX,
T. (77) 74.40.21

44 VERIGNEAUX, 52, rue Coulmiers,
44016 NANTES CEDEX,
T. (40) 74.01.52

45 VOTRE BUREAU, R.N. 744, BP 12,
45015 ORLEANS CEDEX,
T. (38) 91.30.97

49 BURHELIO, 22, rue l'Etanduère,
BP 645, **49006 ANGERS CEDEX,**
T. (41) 88.95.24

51 GL INFORMATIQUE, BP 159,
(Luxémont Vilotte),
51305 VITRY-LE-FRANÇOIS,
T. (26) 74.09.02
PIERRE, BP 194, 2 bis, rue A.-Huet,
51057 REIMS, T. (26) 87.06.44
SERIEEL 7, rue de Chastillon,
51000 CHALONS-SUR-MARNE,
T. (26) 65.43.99

53 MAINE INFORMATIQUE,
route de Mayenne, BP 135,
53007 LAVAL, T. (43) 53.06.11

54 RITTER, 1, place Carnot,
54000 NANCY, T. (8) 332.06.68
COMPUTERLAND, 49, rue du Pont,
54000 NANCY, T. (8) 337.16.65
SEMITEC, 69, rue de Maréville,
54520 LAXOU, T. (8) 340.43.38

57 CENTRE DE MICRO-
INFORMATIQUE, ETS BECKER,
1, Place de la Gare,
57800 FREYMING MERLEBACH,
T. (8) 704.50.57

59 INFORMATIQUE CENTER
17, rue Nicolas-Leblanc,
59000 LILLE, T. (20) 54.61.01
LEANORD, 236, rue Sadi-Carnot,
59320 HAUBOURDIN,
T. (20) 07.30.55
SEROSI, 2, rue de la Créativité,
Cousinerie Sud,
59650 VILLENEUVE-D'ASCQ
T. (20) 05.23.38
SETI 17/19, av. Foch, BP 139,
59306 VALENCIENNES,
T. (27) 45.15.60

60 QUENEUTTE
5, rue du Docteur Gérard,
60000 BEAUVAIS, T. (4) 445.12.74
QUENEUTTE
22, rue de la République,
60108 CREIL, T. (4) 425.04.26

63 ENGETEL, 36, rue Gutenberg,
Z.I. de Brézet,
63039 CLERMONT-FERRAND CEDEX, T. (73) 91.32.42
TECHNICENTRE 2, pl. Salford,
63000 CLERMONT-FERRAND,
T. (73) 92.52.41

66 I.B.S., 14, rue de Vauvenargues,
66000 PERPIGNAN, T. (68) 85.17.11

67 ALSATEL, 2, rue Charles Wurtz,
67000 WOLFISHEIM-STRASBOURG, T. (8) 878.01.56
SMI-PASCAL INFORMAT.
2, quai de Paris,
67000 STRASBOURG, T. (88) 22.11.32

69 COMP/SHOP JANAL
12, cours d'Herbouville,
69004 LYON, T. (7) 839.44.76
SUDETEL, 18, cours Suchet,
69002 LYON, T. (7) 838.15.15
TELEPHONIE GENERALE
79, rue de l'Abondance,
69422 LYON CEDEX 3,
T. (7) 860.15.58
TELEPHONIE LYONNAISE
230, rue du 4-Août,
69615 VILLEURBANNE,
T. (7) 884.45.16

72 SOMEP INFORMATIQUE
39, rue du Docteur LEROY,
72000 LE MANS, Tél. (43) 24.32.67

73 E.T.S./SI-TEL, Z.I. du Cheminet,
73006 CHAMBERY, T. (79) 69.27.27

74 E.T.S./SI-TEL, 26, rue de Chambéry,
74000 ANNECY, T. (50) 45.28.40

75 REGLE A CALCUL
67, bd St-Germain, **75005 PARIS,**
T. 325.68.88
FNAC, 136, rue de Rennes,
75006 PARIS, T. 544.39.12
PIERRE S.A., 36, rue Lafitte,
75009 PARIS, T. 770.46.44
ILLEL, 86, bd Magenta,
75010 PARIS, T. 201.94.68
MID, 47, avenue de la République,
75011 PARIS, T. 357.83.20
TRIANGLE INFORMATIQUE,
64, bd Beaumarchais, **75011 PARIS,**
T. 805.62.00
COMPUTERLAND,
135, bd Voltaire, **75011 PARIS,**
T. 379.21.01
ILLEL, 143, av. Félix-Faure,
75015 PARIS, T. 554.83.81
SIDEG, 125, rue Legendre,
75017 PARIS, T. 627.12.43
SIGMATIQUE 10, rue de Civry,
75016 PARIS, T. 743.04.83

76 SCRIPTA, 27 rue Jeanne-d'Arc,
76000 ROUEN, T. (35) 89.46.39

77 COMPUTER SOFT, BP 28,
77300 FONTAINEBLEAU,
T. 422.25.12

78 TRIANGLE INFORMATIQUE,
2 bis, rue St-Honoré,
78000 VERSAILLES, T. (3) 953.51.63

81 I.T.L., 8, boulevard Soult,
81000 ALBI, T. (63) 54.57.91

83 S.I.A., Le Paillon, av. Brunet,
83100 TOULON, T. (94) 23.74.30

91 CODELEC, BP 90 (Z.I. Courtabœuf),
91402 ORSAY CEDEX, T. 928.01.31



3 STANDS PROCEP
pour vous accueillir quel que soit le but de votre visite.

Pour les fanas de la micro-informatique comme pour les débutants :
SICOB-BOUTIQUE
(stands n° 124, 126 et 128).

Pour les spécialistes du domaine industriel :
SICOB O.E.M.
(stands n° 157 et 159).

Pour les chefs d'entreprise, les cadres et les gestionnaires :
SICOB PALAIS du CNIT
(niveau 3 - stand 3 C 3308)

SFC

PROCEP
19-21, rue Mathurin-Régnier
75015 Paris
Tél. : 306.82.02



COMMODORE LEADER EUROPEEN DE LA MICRO-INFORMATIQUE

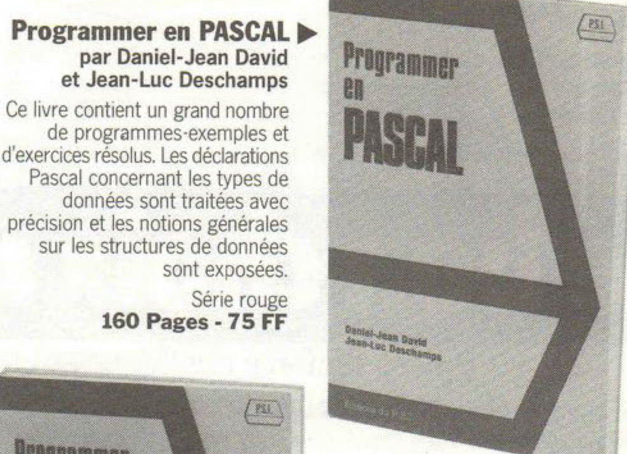
Des livres pour votre CBM



◀ La découverte du PET/CBM
par Daniel-Jean David

Cet ouvrage d'initiation à l'utilisation de la gamme PET/CBM comprend, essentiellement, une introduction progressive au langage Basic. La découverte du langage est conduite en bâtissant des programmes de plus en plus élaborés, au fur et à mesure de l'introduction de notions nouvelles.

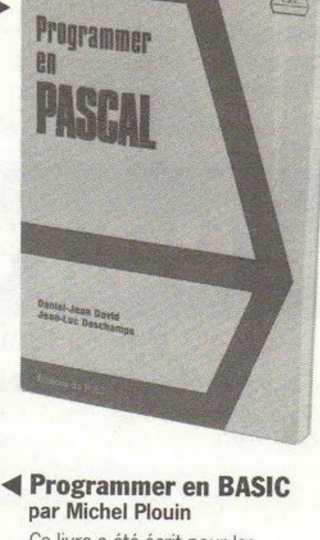
Série verte.
136 Pages - 65 FF

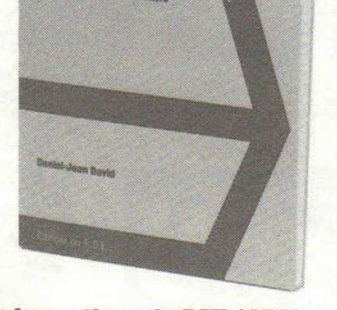


Programmer en PASCAL ▶
par Daniel-Jean David et Jean-Luc Deschamps

Ce livre contient un grand nombre de programmes-exemples et d'exercices résolus. Les déclarations Pascal concernant les types de données sont traitées avec précision et les notions générales sur les structures de données sont exposées.

Série rouge
160 Pages - 75 FF





La pratique du PET/CBM ▶
(Volume I)
par Daniel-Jean David

Cet ouvrage, qui suppose une bonne connaissance du Basic et des commandes du PET/CBM, ouvre les portes des applications faisant appel aux fichiers (cassettes, disquettes) à l'impression et au bus IEEE. Il comporte également de nombreux exemples et exercices avec solution.

Série bleue
136 Pages - 65 FF

◀ Programmer en BASIC
par Michel Plouin

Ce livre a été écrit pour les utilisateurs d'ordinateurs individuels en particulier d'Apple II, TRS-80 et PET/CBM.
Un répertoire Basic rend son utilisation très pratique et facilite la transposition d'un programme écrit pour un P.S.I. sur un autre.

Série verte
132 Pages - 65 FF



◀ La pratique du PET/CBM
(Volume II)
par Daniel-Jean David

Ce volume est une initiation à la programmation en langage machine 6502, dont le jeu d'instruction est expliqué et utilisé. L'assembleur symbolique, l'éditeur et le chargeur y sont décrits. L'interaction avec le Basic et avec le système y sont étudiés.

Série rouge
176 pages
75 FF

Jeux, trucs et comptes ▶
par Michel Benelfoul

Ce livre est destiné à l'amateur de programmation débutant, désorienté par une mémoire vierge de toute instruction. Il y trouvera 29 programmes utilisant un maximum de fonctions du PET/CBM et abordant des domaines très variés, des jeux à la paie et à la facturation.

Série verte
160 pages
75 FF



P.S.I.

EDITIONS DU P.S.I.

BON DE COMMANDE

NOM
RUE
CODE POSTAL
VILLE

CBM 1

DÉSIGNATION	NOMBRE	PRIX
Les prix sont : taxes, emballage et port compris (envoi par avion + 5 FF par ouvrage).	TOTAL	

Envoyer ce bon accompagné de votre règlement à ÉDITIONS DU P.S.I.
41-51, rue Jacquard BP 86
77400 Lagny s/Marne Tél. (6) 007.59.31.

```
*********************************
*                               *
* LA NOUVELLE GAMME COMMODORE *
*                               *
*********************************
```

Cette année a été fertile en évolution dans la gamme des ordinateurs Commodore: Le CBM 8032 s'est affirmé. Les 4016 et 4032 sont venus remplacer les 3016 et 3032, en leur apportant les avantages du BASIC 4.0. L'unité de disques 4040 est venue remplacer la 3040, apportant la possibilité de fichiers relatifs. L'imprimante 4022 remplace la 3022, pendant que, dans la série 8000, les 8026 et 8027 viennent à coté des 8024 apporter la qualité d'impression "courrier". Nous apprenions ensuite l'imminente disparition du bon vieux P.E.T.2001 et l'arrivée du VIC destiné à prendre le marché bas de gamme.

Comme le SICOB 81 doit être le cadre d'annonces - ou de début de commercialisation - importantes, nous pensons qu' il est utile de faire le point, d'autant que courent certains bruits sur des incompatibilités, des différences etc... Ces bruits n'ont pour cause que l'ignorance de ceux qui les font courir et nous pensons qu'il est essentiel de les dissiper. Vous n'avez aucune inquiétude à avoir: la gamme Commodore est une des plus étendues du marché, et en même temps une des plus compatibles.

Revenons un instant à l'état du dernier SICOB. La gamme est formée du P.E.T. 2001, du système 3001: 3032 (25 lignes de 40 caractères chacune, clavier professionnel), 3040 (double disquette 340 Ko), imprimante 3022 (80 col., 60 car./s) et du système 8001: 8032 (25 lignes x 80 caractères, clavier encore plus semblable à celui d'une machine à écrire, BASIC 4.0), 8050 (double disquette

1 Mo), imprimante 8024 (132 colonnes, 170 car./s). A côté du 3032, on trouve les tailles mémoire réduites 3008 et 3016.

Il y a différentes compatibilités le long de la gamme: par exemple, sauf pour ceux qui ont été livrés dans les premiers, le P.E.T. 2001 est en effet un 3008: il a la même carte et les mêmes mémoires mortes; on peut lui connecter des disques. Il y a ensuite une compatibilité de connexion sur le bus IEEE qui vous permet de connecter un périphérique 8000 sur une unité centrale 3000 et vice-versa: vous pouvez connecter une 8024 ou une 8050 sur un 3032, une 3040 sur un 8032. Vous pouvez connecter plusieurs unités de disques sur la même unité centrale, et elles peuvent être différentes (une 8050 + une 3040 par exemple).

Au point de vue logiciel, il y a quelques différences entre périphériques homologues. Par exemple l'imprimante 8024 n'a que les adresses secondaires 0,1,2 et 3: elle n'a pas les possibilités graphiques de la 3022 ce qui est normal pour une imprimante utilisée plutôt en gestion.

Du côté des disques les différences sont plus grandes: il n'y a pas compatibilité physique en ce sens que, vu que la densité des pistes n'est pas la même, une disquette créée sur 3040 n'est pas lisible sur 8050 et réciproquement. Si vous passez de 3040 à 8050, il faut pendant un temps avoir les deux unités connectées et copier vos fichiers de l'une à l'autre (au contraire des copies d'une platine à l'autre dans une même unité, cette copie passe par l'unité centrale). En revanche, et c'est très important, toutes les commandes de la 3040 sont reconnues par la 8050 et, par suite , vos programmes de fichiers qui tournaient sur 3040 doivent tourner sur 8050, et inversement. Les seules adaptations nécessaires concerneront des accès explicites à des numéros de piste et secteur: la piste du catalogue et le nombre de secteurs par piste sont différents. Il sont résumés dans le tableau 1 à la fin de cet article.

Une autre différence au niveau des disques, et qui est en correspondance avec des différences au niveau du BASIC est que:

1) La 8050 admet des commandes supplémentaires par rapport à la 3040; les plus importantes étant:

a) l'initialisation automatique

b) l'ajout à la fin des fichiers séquentiels

et surtout c) l'accès relatif.

2) Par rapport au BASIC du 3032 (BASIC 2.0, il n'y a pas eu de BASIC 3.0 diffusé), le BASIC 4.0 introduit deux changements.

a)Une amélioration spectaculaire de la routine de "garbage collection" (compactage des chaînes de caractères qui ne sont plus utilisées), et surtout

b) de nouvelles commandes disque, comme DIRECTORY, HEADER, COPY etc.. et la variable réservée d'état disque DS et DS$.

Ces commandes évitent d'ouvrir le canal 15 et d'envoyer une chaîne de commande: ainsi COPY "ZOZO",D1 TO "ZOZO",D0 est équivalent à

OPEN 15,8,15:PRINT#15,"C0:ZOZO=1:ZOZO"

Mais l'ancienne méthode marche toujours, c'est pourquoi vos anciens programmes de 3000 doivent tourner sur votre 8000.

De fait, lorsque BASIC 4.0 rencontre un nouvel ordre disque, il le transforme en la chaîne de caractères correspondante. Il est de même pour les ordres BASIC 4.0 qui font appel aux nouvelles possibilités comme l'accès relatif. Ainsi à RECORD correspond PRINT#15,"P..."

Par suite, si vous connectez une 8050 à un 3032, vous pourrez exploiter ses nouvelles possibilités même en BASIC 2.0. Beaucoup d'utilisateurs ne le savent pas.

Ces points étant clarifiés, passons à la nouvelle gamme.

****
* LA SERIE 4000 *
****

Cette série vient remplacer la série 3000 au niveau unité centrale (4016 et 4032), unité de disques (4040) et imprimante (4022).

L'imprimante 4022 est une EPSON comme la 3022, dont elle a toutes les possibilités, avec 3 adresses secondaires supplémentaires. Elle est plus silencieuse, plus compacte et a une meilleure qualité d'impression. Sinon, il y a parfaite compatibilité.

Au niveau unité centrale, le 40xx est un 30xx avec de nouvelles MEM qui fournissent BASIC 4.0. L'écran est en 40 colonnes. La carte interne est la même. Ce sont les boîtiers MEM qui sont changés. Noter qu'il y en a un de plus: de B000 à BFFF. Il y a donc un emplacement libre en moins. Vous pouvez transformer un 3000 en 4000 par changement des MEM. On obtient une machine intermédiaire entre le 3000 et le 8000: on a BASIC 4.0, mais les nouvelles possibilités d'édition du 8000 dues à son nouveau clavier et à l'écran 80 colonnes sont absentes.

Noter qu'il doit apparaître un nouveau modèle de 4000 avec écran de la même taille qu'un 8000 mais toujours 40 colonnes par ligne: les caractères seront donc plus gros et lisibles de loin (intéressant dans une classe). Noter aussi que le 4008 n'est pas diffusé en France: il ne serait pas beaucoup moins cher que le 4016.

Au point de vue compatibilité, les programmes purement BASIC doivent pouvoir passer tels quels sur 3000, 4000 et 8000 (à condition de ne pas avoir la variable DS ou DS$ ce qui est très rare).

Ceux qui font appel à des routines langage machine situées dans les MEM BASIC doivent être adaptés car toutes les adresses MEM système sont changées (à cause des 4 K supplémentaires). En revanche, les adresses stratégiques de la page zéro sont très peu changées au passage 3000/4000/8000. On rappelle que lors du passage ancien 2001/nouveau 2001-3000, la page zéro avait été complétement bouleversée et les adresses BASIC avaient été légèrement changées.

Du coté disque, on peut dire que la 4040 est une 8050 qui a en gros la capacité de la 3040. Le SED est celui de la 8050: il fait 16K, a en partie l'autoinitialisation et il permet les fichiers en accès relatif (nous consacrerons un article complet à l'accès relatif dans un prochain numéro). A cause d'un formattage différent, la capacité est un peu inférieure à la 3040.

Point qui mérite d'être clarifié: une disquette 3040 peut être lue sur une 4040; une disquette 4040 peut être lue sur une 3040. En revanche, une disquette formatée et commencée à être écrite sur une unité ne peut pas être écrite sur l'autre. Dans le sens 3040 vers 4040, la 4040 émet un diagnostic. Dans le sens 4040 vers 3040 c'est plus insidieux: la 3040 corrompt la disquette qui ne pourra plus être lue. La marche à suivre pour utiliser vos anciennes disquettes 3040 sur votre nouvelle 4040 est la suivante: vous formatez une disquette neuve sur votre 4040, puis vous copiez tous les fichiers de l'ancienne disquette par COPY D0 TO D1. Il faut que votre disquette ne soit pas totalement pleine puisque la 4040 a une capa-

cité légèrement diminuée.

Tableau 1 - Pistes et secteurs sur 3040/4040/8050

modèle	3040	4040	8050
catalogue et BAM	piste 18 secteur 0	piste 18 sect 0 et 1	piste 39 sect 0 et 1 P 38 sect 0 à 3

piste	sect. (nb)		sect (nb)		piste	. sect. (nb)	
1-17	0-20	21	0-20	21	1-39	. 0-28	29
18-24	0-19	20	0-18	19	40-53	. 0-26	27
25-30	0-17	18	0-17	18	54-64	. 0-24	25
31-35	0-16	17	0-16	17	65-77	. 0-22	23

En résumé, il y a sur la gamme une compatibilité assez complète et le passage 3000/4000 devrait se faire en douceur. Notons que l'EDEX 4000 ne devrait pas tarder à apparaître.

Voyons maintenant les nouveautés du SICOB 81.

* LE VIC *

Le PET 2001 est en train de disparaître. Le bas de la gamme Commodore sera maintenant constitué par le VIC qui est un boîtier contenant un clavier professionnel et qui se connecte à un téléviseur couleur. Voyez l'article qui lui est consacré dans ce numéro. Il faut noter que le VIC s'intègre bien à la famille Commodore: on peut lui adjoindre une interface IEEE (disponible prochainement) qui permettra la connexion des périphériques de la gamme. Les adresses des routines système sont changées ainsi que les adresses page zéro, mais on retrouve les mêmes concepts et les mêmes techniques. Donc, là aussi, compatibilité.

****
* LE 8096 *
****

Il s'agit tout simplement d'un 8032 auquel on ajoute une carte de 64 K de mémoire vive. Bien sûr, toute la mémoire n'est pas accessible au même moment: la MEV est organisée en blocs de 16 K: seuls deux blocs sont accessibles à un instant donné. C'est géré par un registre qui indique par un bit 1 quels blocs sont actifs. Le dispositif peut se rajouter à un 8032 acheté précédemment. Il y a parfaite compatibilité logicielle puisqu'un des états est celui où les 32 K habituels sont actifs.

****
* MICRO-MAINFRAME *
****

Le haut de la gamme est formé par le CBM 9000 MICRO-MAINFRAME. Il résulte d'une opération très intelligente. Les gens de l'université de Waterloo avaient conçu un système très perfectionné (très riche en interpréteurs de différents langages) à base de 6809. Ils avaient besoin d'un clavier-écran-disque bon marché: un CBM 8032 / 8050 le fournit.

Le Micromainframe est en fait un 8032 + 8050 ayant une carte 6809 / 64 K MEV à l'intérieur. Un interpréteur permet de choisir entre le mode 6502 où l'on dispose du Basic du CBM, de l'assembleur 6502, de Pascal Transam, de Comal, Forth, Lisp...et le mode 6809 où l'on dispose du Basic Waterloo qui est plus performant que le Microsoft, de Fortran, d'APL, d'un Pascal étendu et où on aura bientot Cobol. C'est l'outil idéal en enseignement où l'on veut enseigner le plus possible de langages différents.

Bien sûr, nous vous en dirons plus dès que nous aurons pu évaluer le produit.

****
* IMPRIMANTES *
****

Terminons en signalant que, dans la série 8000, la 8024 est maintenant épaulée par deux imprimantes à marguerite la 8026 (qui a un clavier) et la 8027 qui procure la qualité "courrier" à un prix très modique. Ces imprimantes sont prévues pour utilisation avec la version française du système de traitement de textes avec le clavier AZERTY accentué en option sur le 8032.

Daniel-Jean DAVID

```
***********************
*                     *
* VISITE AU PET SHOW *
*                     *
***********************
```

Tous les ans, vers la fin juin, à Londres, se tient le PET show. De quoi s'agit-il? Tout simplement d'une exposition qui a l'importance de la SICOB Boutique ou - à peu près- de la MICROEXPO, mais qui est entièrement consacrée au PET / CBM et aux produits compatibles. Ceci est une preuve de la domination qu'exercent les produits Commodore sur le marché d'Outre-Manche.

Nous avons eu le privilège de visiter l'exposition cette année qui est particulièrement riche en annonces Commodore. Nous tenons à faire part de nos impressions à nos lecteurs, en avant-première des produits qui seront présentés au SICOB.

Nous avons d'abord été frappé par la fréquentation: nous avons visité le PET show le dernier jour, un samedi, par un très beau temps. Eh bien, la foule se pressait aux stands et, alors que la fermeture était prévue pour 17 heures, des affaires se concluaient encore dans la fièvre à 18 heures!

Ensuite, nous avons remarqué le professionnalisme et le sérieux des produits présentés. Il y avait peu de gadgets, par exemple un stylo lumineux (light-pen: nous vous apprendrons comment en bricoler un dans le prochain numéro). Sinon,les matériels et les logiciels présentés étaient spectaculaires et nous espérons que beaucoup viendront en France.

Au point de vue matériel, nous avons particulièrement remarqué:

-plusieurs cartes haute résolution
-plusieurs interfaces RS232 accompagnées du logiciel permettant de transformer le CBM en terminal performant.

-les programmes de gestion sont inutilisables en France à cause des différences de législation. Toutefois, certains des logiciels présentés étaient remarquables.

-Un système nous a semblé très spectaculaire, c'est le Silicon Office de Bristol Software. Il intègre sur un CBM 8096 les fonctions de mailling, paie, gestion de base de données comme OZZ, VisiCalc, traitement de texte etc... Comme son nom l'indique, un véritable bureau en silicium!

Le stand Commodore occupait une surface importante. Il présentait beaucoup de nouveautés:

-le VIC, bien sûr. Tous les VIC exposés étaient occupés par des jeunes en train de jouer. J'ai pu apercevoir l'unité monodisquette destinée au VIC ainsi que son magnétophone: il est comme celui que vous connaissez pour le CBM, à part qu'il est blanc... et qu'il a un compteur!

-le 8096. Etant donné ce qu'on peut faire avec un CBM 8032, que ne fera-t'on avec 64 K de MEV en plus?

-le CBM 9000 MICRO-MAINFRAME, formidable produit pour l'éducation vu le nombre de langages de programmation qu'il possède sur ses deux microprocesseurs: BASIC, Assembleur, PASCAL, LISP, FORTH, FORTRAN, APL et bientôt COBOL.

Une bonne partie du stand était réservée à la démonstration de logiciels pour l'enseignement. Cela nous a permis d'essayer le MTC-PILOT qui est un langage d'écriture de dialogues machine-élève pour les professeurs non informaticiens.

Une évaluation détaillée en sera donnée dans un prochain numéro de La Commode.

Enfin, pour la première fois, PROCEP avait un stand important pour préparer l'exportation des produits français les plus remarquables développés sur CBM. Citons:

-EDEX dans ses différentes versions 3000 et 8000. Vous savez tous qu'EDEX est une MEM qui permet d'ajouter à BASIC des commandes ou instructions comme IF THEN ELSE, PRINTUSING, APPEND/MERGE ou l'exécution pas à pas.

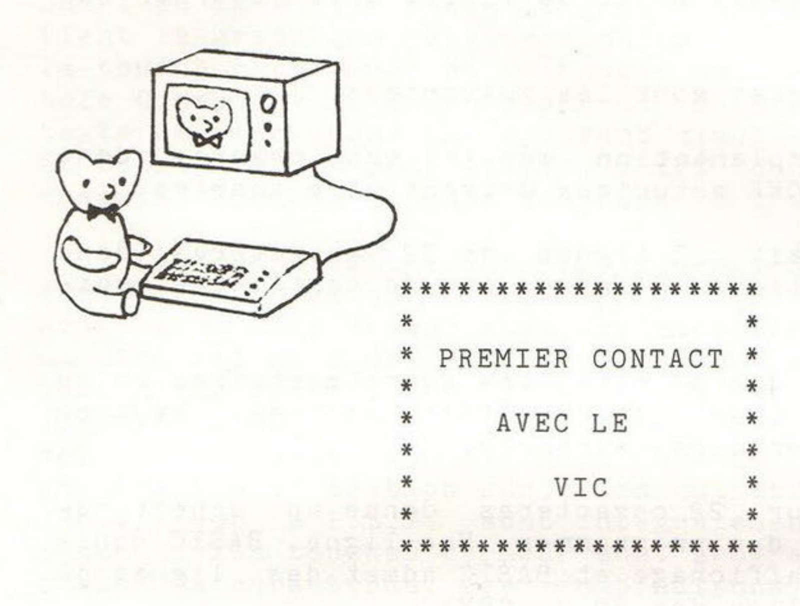
-MULTEX, un logiciel de gestion de l'IEEE qui permet de relier plusieurs CBM à une même unité de disques et ainsi interroger une base de données à partir de plusieurs postes.

-MASTER, un logiciel d'acquisition de données sur écran avec gestion de masques à l'aide d'instructions ajoutées au BASIC.

Bien entendu, tous ces produits seront évalués en détail dans La Commode.

En conclusion, une exposition à ne pas manquer, en attendant qu'il y ait un PET show français.

Daniel-Jean DAVID



PREMIER CONTACT AVEC LE VIC

Le VIC a été le clou de la dernière NCC à Chicago. Il est vrai qu'offrir tant de puissance de calcul et la couleur pour 2500 F tient du prodige.

Nous offrons ici nos premières impressions. Nous publierons bien sûr d'autres études plus approfondies à mesure que nous connaîtrons mieux le produit.

Le VIC est destiné à remplacer le P.E.T. vers le bas de gamme. Or ce sont les anciens utilisateurs du P.E.T. qui seront surpris avec le VIC: en effet, le caractère monolithique du P.E.T. est abandonné. Le VIC est un boîtier-clavier/unité centrale qui se connecte à un téléviseur extérieur. Il y a même un chassé-croisé avec TANDY dont le modèle I -en éléments séparés- s'opposait au P.E.T. monolithique, alors que maintenant, c'est le TANDY modèle III qui est en un seul coffret alors que le VIC est en éléments séparés. Il faut toutefois noter que le VIC offre la couleur: il est donc plus normal que le té-

léviseur soit séparé (et non compris dans le prix; il en est de même pour le TANDY Color Computer).

Cette première surprise passée, on se retrouve en pays de connaissance: le VIC a EXACTEMENT le même BASIC que le P.E.T. ou le CBM, les mêmes ordres, les mêmes fonctions. Un programme BASIC de P.E.T. doit tourner sur le VIC.

Les seules différences sont les suivantes:

1-la carte d'implantation mémoire est changée, donc les programmes à POKE astucieux doivent être adaptés.

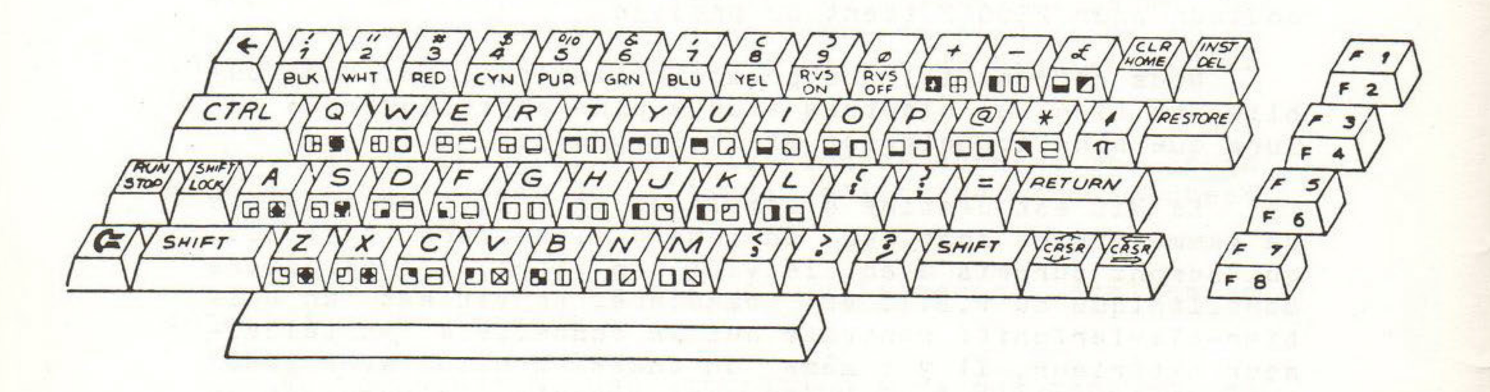
2-l'affichage fait 23 lignes de 22 caractères; donc certaines instructions d'impression doivent être modifiées.

3-pour profiter des possibilités supplémentaires en ce qui concerne la couleur et les effets sonores, tout se passe par POKE à certaines adresses.

L'affichage sur 22 caractères donne un aspect curieux aux listings des programmes. Une ligne BASIC équivaut à 4 lignes d'affichage et BASIC admet des lignes de 88 caractères (au lieu des 80 du CBM).

****
* LE CLAVIER *
****

L'organisation du clavier est montrée par la figure ci-dessous.



Il faut noter que ce clavier est de qualité professionnelle parfaite, tant pour la taille des touches que le toucher. On retrouve la plupart des touches du CBM, à

l'emplacement près. Les touches lettres ont 4 valeurs: lettre majuscule, lettre minuscule, et les deux graphiques sur l'avant.

A la mise sous tension, on est en mode graphique: l'appui sur la touche seule donne la lettre majuscule; SHIFT et la touche donne le graphique de droite; on obtient le graphique de gauche en tapant simultanément sur la touche considérée et la touche qui représente le symbole Commodore C=. On passe du mode graphique au mode texte et vice-versa en appuyant simultanément sur SHIFT et Commodore. En mode texte, SHIFT donne la lettre minuscule, Commodore donne toujours le graphique de gauche.

La touche CTRL (Contrôle) fournit des caractères de contrôle, notamment avec les touches couleurs: après avoir fait CTRL "bleu" tous les caractères tapés viennent en bleu (il ne faut pas que le fond soit bleu si vous voulez les voir!). Les touches RUN/STOP, RVS ON et OFF (qui s'obtiennent avec CTRL), CLR/HOME, INST/DEL et curseur jouent exactement le même rôle que sur un P.E.T. Les possibilités d'édition sur écran qui étaient un avantage décisif sur le P.E.T. sont intégralement conservées. De même que les touches curseur sont programmables dans une chaîne de caractères, les combinaisons de changement de couleur le sont.

RESTORE rétablit le VIC dans son état de mise sous tension, mais le programme est conservé en mémoire: c'est un reset tiède. Enfin, il y a deux fois (avec SHIFT) quatre touches dont le rôle est programmable par l'utilisateur à la droite du clavier.

Notons que, sur le VIC, si à INPUT A on répond par RETURN tout court, alors on ne sort pas du programme comme avec le P.E.T.: la variable A garde son ancienne valeur.

****
* INTERFACES *
****

Le VIC comporte de nombreux connecteurs vers l'extérieur.

1- Le connecteur vers la télévision fera l'objet de la section suivante.

2- Il y a un connecteur magnétophone identique à celui

du P.E.T. Une cassette de P.E.T. est lisible sur le VIC, ce qui est un bon facteur d'intégration du VIC au reste de la gamme.

3- Un connecteur appelé connecteur de contrôle permet de relier des "manches à balais".

4- Le connecteur IEEE est très particulier: en effet les signaux de l'IEEE sont transmis en série, et non en parallèle, sur le VIC. Donc le logiciel obéit aux mêmes principes que sur le P.E.T. et moyennant un interface prévue par Commodore pour reconstituer l'IEEE classique, tous les périphériques de la gamme CBM et les appareils d'instrumentation habituels seront connectables au VIC.

5- Un connecteur d'extension-mémoire transmet les bus du microprocesseur. On peut y connecter des cartouches, dont la première disponible sera une extension de 3 K de MEV portant la MEV totale à 8 K. D'autres extensions de MEM ou de MEV sont prévues jusqu'à occuper tout l'espace adressable de 64 K.

****
* LE PROBLEME DE LA TELEVISION *
****

Le VIC (Video Interface Computer) commande un téléviseur couleur et aussi le haut-parleur de ce téléviseur (3 voies son + 1 voie de bruitage) grâce à un circuit intégré très évolué qui s'appelle aussi un VIC (Video Interface Chip). Ce boîtier existe en deux versions : PAL et NTSC, donc aussi le microordinateur VIC. Au départ, c'est des VIC PAL qu'on aura en France.

Cela pose un problème en France. Il faut

ou bien 1) développer une interface PAL-SECAM qui peut être coûteuse;

ou bien 2) acheter en même temps un téléviseur PAL ; on perd l'avantage du prix qu'on avait en utilisant le téléviseur domestique;

ou bien 3) se contenter d'un moniteur noir et blanc, mais on perd l'avantage des effets couleur;

Nous pensons qu'en définitive la meilleure solution est 2. En effet, se connecter au téléviseur familial n'est pas une très bonne solution: l'activité microinfor-

matique prive le reste de la famille de la télévision. D'autre part le téléviseur familial a généralement un écran très grand, peu pratique à regarder de près pour visualiser des textes. De fait, l'expérience montre que, partout où on a connecté un microordinateur au téléviseur familial, il y a peu de temps après, achat d'un second téléviseur!

Il nous semble donc tout-à-fait à conseiller d'acheter un téléviseur multistandard PAL/SECAM portatif, qu'on devrait pouvoir trouver pas trop cher et qui pourra servir de téléviseur d'appoint pendant les vacances.

****
* IMPLANTATION MEMOIRE *
****

Si l'on retrouve les mêmes notations que sur le P.E.T ., les routines système sont déplacées et les adresses système de la page zéro ont été déplacées elles aussi. Notons qu'il n'y a pas de moniteur langage machine. Celui-ci sera dans une extension MEM.

Le premier K de mémoire est la MEV système de 0000 à $3FF. Le texte BASIC réside de $1000 (4096 en décimal) à $1DFF normalement, de $400 (1024) à $1DFF en présence de l'extension de 3 K MEV en cartouche.

De $1E00 (7680) à $1FFF on trouve la MEV écran (512 cases) pour les codes caractères.

De $9600 (38400) à $9800, il y a une autre MEV écran pour le code couleur de chaque caractère.

Le générateur de caractères est dans l'espace adressable, de $8000 (32768) à $8FFF. Le bloc $9000-$9FFF non consacré à la MEV couleur est consacré aux entrées-sorties. Le boîtier VIC est en $9000. De C000 à FFFF on trouve la MEM système, avec l'interpréteur BASIC. Les autres zones sont libres pour des extensions MEV ou MEM de $2000 à $7FFF et de $A000 à $BFFF.

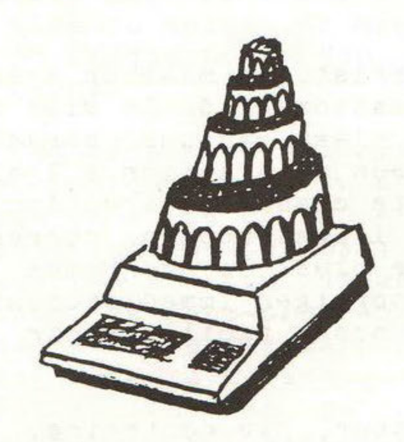
Nous donnons au verso quelques adresses de page 0, 1, 2 et 3 qui ont été repérées. Il y en aura d'autres dans nos prochains numéros:

VIC (dec/hexa)	CBM (dec/hexa)	Rôle
43-44/2B-2C	40-41/28-29	début du texte BASIC
45-46/2D-2E	42-43/2A-2B	début des variables
47-48/2F-30	44-45/2C-2D	début des tableaux
49-50/31-32	46-47/2E-2F	fin des tableaux
55-56/37-38	52-53/34-35	fin de la mémoire
97-102/61-66	94-99/5E-63	accu. flottant no 1
122-123/7A-7B	119-120/77-78	pointeur vers le texte BASIC dans la routine CHRGET
144/90	150/96	variable ST
158/9E	192/C0	compteur d'erreur bande
198/C6	158/9E	pointeur dans le tampon clavier
204/CC	167/A7	0 = curseur clignote 1 = ne clignote pas
209-210/D1-D2	196-197/C4-C5	pointeur vers le début de la ligne écran
211/D3	198/C6	numéro de la colonne où est le curseur
214/D6	216/D8	no de ligne du curseur
243-244/F3-F4	pas d'équivalent	pointeur vers le code couleur du caractère courant.
631/277	623/26F	début tampon clavier
646/286	/	couleur courante
649/289	/	longueur max. du tampon clavier de 10 à 80.
650/28A	/	si = 128, toutes les touches ont la répétition
788-789/314-315	144-145/90-91	vecteur d'interruption.

On voit que beaucoup d'adresses n'ont fait que d'être repoussées de 3 octets, mais leur rôle est identique à celui sur le PET/CBM.

Pierre-Etienne THALBERG

```
****************************************
*                                      *
* LE C.B.M., UN ORDINATEUR POLYGLOTTE *
*                                      *
****************************************
```



Bien que 90% des utilisations du CBM se fassent en BASIC, celui-ci permet maintenant quantité d'autres langages tous utiles chacun dans son domaine d'applications.

```
*********
* BASIC *
*********
```

BASIC reste le roi des langages pour microordinateurs. Au départ, BASIC a été défini comme un sous-ensemble simplifié de FORTRAN qui était à l'époque le langage de programmation scientifique le plus répandu. Cette simplification avait pour but d'obtenir un langage plus facile à apprendre quitte à accepter certaines limitations. De fait, le but a été parfaitement atteint: BASIC est le langage le plus facile à apprendre: il suffit de quelques heures passées devant votre ordinateur pour savoir BASIC.

Maintenant, les limitations de BASIC, qu'on était prêt à accepter pour payer la simplicité, ne sont pas si importantes que cela. En fait, BASIC a été progressivement étendu, et le BASIC du CBM est un des plus étendus, surtout si l'on tient compte d'EDEX, de sorte qu'il est maintenant un langage évolué très puissant. Il est aussi puissant que FORTRAN, et même plus sous certains aspects. En particulier, BASIC est doté de facilités de traitements de chaînes de caractères à la fois puissantes et commodes d'emploi que tous les langages peuvent lui envier. Ces facilités autorisent, comme l'expérience le

montre, des traitements de gestion. Couplées à certaines caractéristiques des disques COMMODORE, on a le moyen de manipuler les noms de fichiers à l'intérieur des programmes, ce que très peu de systèmes permettent.

La seule faiblesse du langage Basic qui reste,est sa manière rudimentaire de traiter les sous-programmes: pas d'appel par un nom symbolique, pas de transmission d'arguments.

Une autre caractéristique mise en avant pour BASIC est le caractère conversationnel de la mise au point des programmes. En fait, ce n'est pas une caractéristique du langage BASIC, mais de son utilisation à l'aide d'un interpréteur qui interprète chaque instruction une par une: donc, en cas d'erreur, l'instruction correspondante est facilement repérable. De plus, le programme est toujours en mémoire, on peut le corriger immédiatement.

L'opération est encore facilitée par les possibilités d'édition du CBM.

Avec un compilateur, au contraire, le programme est traduit en bloc en langage machine et ensuite seulement, le programme traduit est exécuté. Les erreurs sont plus difficiles à repérer et, pour faire une correction, il faut recharger le programme en mémoire, corriger, puis tout recompiler. En revanche, on y gagne en taille mémoire et surtout en vitesse d'exécution (des facteurs de 10 à 20).

Une erreur souvent commise est de dire que tel ou tel langage est meilleur parce qu'il est interprété ou compilé. Ce n'est pas une caractéristique du langage et donc, on ne doit pas en faire un critère de choix du langage. Un même langage peut d'ailleurs posséder à la fois des interpréteurs et des compilateurs, c'est le cas de la plupart des langages et notamment de BASIC sur CBM: on dispose de l'interpréteur bien connu et ensuite, il existe un compilateur fait par DTL que nous analyserons dans un prochain numéro. Ceci représente la solution idéale: on met au point le programme avec l'interpréteur puis on compile. Les exécutions se feront à plus grande vitesse avec le programme compilé.

* ASSEMBLEUR *

Si le compilateur, pourtant plus rapide que l'interpréteur,ne procure pas de performances suffisantes alors il faut avoir recours au langage machine du micro-

processeur. De très petits programmes peuvent être entrés dans la machine à l'aide du moniteur langage machine incorporé en MEM. Si le programme est plus important, on ne peut se passer d'un assembleur symbolique, programme qui lit votre texte source sous forme mnémonique (avec instructions et variables symboliques, exemple LDA TOTO) et le traduit en langage binaire ou hexadécimal, un peu à la manière d'un compilateur. Il existe de nombreux assembleurs symboliques sur PET/CBM, de performances variables, exigeant plus ou moins de mémoire et fonctionnant avec cassettes ou disquettes. Nous en présenterons prochainement une revue.

Mais le CBM propose des langages autres que BASIC ou l'assembleur. Pourquoi vouloir un autre langage évolué que BASIC, puisqu'on a le choix entre trois niveaux de performance: BASIC interprété, BASIC compilé et l'assembleur ? C'est que BASIC à tout de même quelques faiblesses: son mécanisme de sous-programmes est rudimentaire. De plus BASIC ne favorise pas la programmation structurée et même, il a les défauts de ses qualités: sa trop grande facilité d'emploi incite à se mettre directement à programmer au clavier sans réfléchir d'avance à la conception de son programme.

****
* PASCAL *
****

C'est pourquoi on préconise l'emploi de langages structurés comme PASCAL qui est le deuxième langage derrière BASIC dans l'ordre de fréquence d'emploi sur micro-ordinateurs. On peut dire que PASCAL est plus puissant que BASIC:

-mécanisme d'appel de procédures perfectionné
-programmation structurée
-structures de blocs
-notions très évoluées comme la récursivité
-les types de données définissables par l'utilisateur
etc...

Comme le montre l'article de J-L DESCHAMPS dans ce numéro, le PASCAL du CBM est perfectionné, agréable d'emploi et il ne rompt pas avec les habitudes système du CBM. En revanche, PASCAL n'est pas aussi simple que BASIC, pour les débutants, bien qu'il ait un grand succès en université pour les bonnes habitudes de programmation qu'il inculque.

****
* COMAL *
****

Il vient d'apparaître un langage qui semble cumuler les avantages de BASIC et de PASCAL: c'est COMAL. COMAL est un intermédiaire entre BASIC et PASCAL.

-On peut le considérer comme un sur-ensemble de BASIC dont il conserve la simplicité et auquel il ajoute principalement des instructions de programmation structurée et un mécanisme de sous-programmes évolué.

-On peut aussi le considérer comme un PASCAL sans la gestion des types.

COMAL est présenté plus en détail dans ce numéro. Il est possible qu'il suppplante BASIC.

D'autre part, le CBM vous permet de vous exercer aussi sur deux langages plus marginaux, plus éloignés des autres langages. Ces langages sont moins algorithmiques et chacun est attaché à une structure particulière de données. Il s'agit de LISP et FORTH.

****
* LISP *
****

LISP est un langage de traitement des listes. Beaucoup de problèmes, notamment d'intelligence artificielle, sont traitables en LISP beaucoup plus facilement qu'avec d'autres langages.

****
* FORTH *
****

FORTH est articulé autour d'une structure de pile. Les opérations doivent lui être fournies en notation polonaise. On peut inclure dans les traitements des passages en assembleur symbolique (mais fournis en notation polonaise). Mais la caractéristique la plus importante de FORTH est qu'il est auto-extensible: l'utilisateur peut ajouter ses propres mots-clefs. Il suffit qu'il fournisse la définition du traitement correspondant; cette définition peut inclure de l'assembleur. On voit la richesse des possibilités que cela ouvre.

****
* PILOT *
****

Enfin, on peut utiliser sur CBM le langage PILOT. Ce langage est ce que l'on appelle un langage auteur, c'est-à-dire un langage permettant à un enseignant non informaticien d'écrire des dialogues élève-machine.

La version de MTC est efficace et nous en publierons prochainement un banc d'essai.

$$$$$$$$$$$$$$
$ CONCLUSION $
$$$$$$$$$$$$$$

La liste des langages accessibles sur CBM est impressionante: BASIC interprété et compilé, assembleur, PASCAL, COMAL, LISP, FORTH, PILOT. Elle s'allonge encore sur le CBM 9000 Micro-Mainframe qui dispose en plus de FORTRAN, APL et COBOL (prochainement).

Nous demandons aux lecteurs de nous envoyer leurs évaluations des différents langages,les particularités de chacun de ces langages qu'ils exploitent plus spécialement dans leurs applications afin d'aider d'autres lecteurs à mieux choisir le langage qui leur donnera le plus de satisfaction.

Pierre Etienne THALBERG



```
************************************
*                                  *
* COMAL, LE DERNIER-NE DES LANGAGES *
*                                  *
************************************
```

La plus grande nouveauté logicielle au SICOB 81 sera certainement COMAL. Il s'agit d'un nouveau langage (COMmon Algorithmic Language) inventé au Danemark par Borge Christensen. Il se trouve que la première version en fonctionnement est pour le CBM 4000 ou 8000. Commodore met le langage dans le domaine public, donc il sera fourni gratuitement (on ne paiera que la disquette). COMAL est fourni sous forme d'interpréteur qui occupe 28 K. Cela laisse assez de place sur un CBM 32 K car les programmes peuvent être organisés en recouvrement.

La caractéristique essentielle de COMAL est qu'il constitue en quelque sorte un intermédiaire entre BASIC et PASCAL. On peut dire que COMAL est un BASIC structuré, ou un PASCAL sans les types de données définis par l'utilisateur.

COMAL est appelé à une large audience dans l'enseignement: en effet, comme PASCAL, il introduit les structures de la programmation structurée et il comble parfaitement la lacune essentielle de BASIC dans le domaine des sous-programmes, mais il conserve la simplicité de BASIC et sa facilité d'emploi et d'apprentissage.

La plupart des instructions et des fonctions de BASIC sont conservées. Les extensions SYS (appel d'un sous-programme en langage machine), PEEK et POKE sont conservées. Au lieu d'ASC, on utilise ORD comme en PASCAL et, comme en PASCAL, on distingue les identificateurs sur plus de deux caractères et l'affectation s'écrit := .Mais si vous écrivez A=B, le système corrigera automatiquement en A:=B.

Programmation structurée

COMAL ajoute à BASIC les constructions classiques de la programmation structurée.

Le IF THEN ELSE est introduit sous la forme: IF...THEN...ELIF...ELSE...ENDIF qui est en fait légèrement plus facile à manipuler que les ; en PASCAL. La clause ELIF est exactement la même que ELSIF de ADA (sinon si...).

Les boucles sont:

WHILE...DO...ENDWHILE	(tant que...faire...)
REPEAT...UNTIL	(répéter...jusqu'à...)
FOR...DO...NEXT	(qui ne diffère de BASIC que par la présence de DO)

Enfin on dispose de CASE avec clause OTHERWISE que tous les PASCAL n'ont pas: CASE...OF...OTHERWISE...ENDCASE

Le fait que chacune de ces constructions se termine par un END approprié est beaucoup plus facile à appréhender que les BEGIN...END et ; de PASCAL.

Les sous-programmes

COMAL dispose d'un mécanisme de sous-programmes appelables par nom symbolique et capables de transmettre des arguments. Ce sont les procédures. Une procédure est définie par:

```
PROC nom(arguments)
...
...
ENDPROC nom
```

et elle est appelable par EXEC nom (il n'y a plus de GOSUB).

Au niveau des arguments, la gestion est aussi perfectionnée qu'en PASCAL puisqu'on dispose de deux sortes d'arguments, les arguments simples qui sont copiés dans le sous-programme (transmission par valeur), et les arguments REF qui sont gérés comme les arguments VAR de PASCAL, avec transmission par adresse:

```
PROC TOTO(A,B,REF X)
```

Lorsqu'un tableau est transmis (par REF) on le fait figurer avec des dimensions vides:

```
PROC TOTO(REF MAT( , ))
```

On peut rendre locales toutes les variables d'une procédure en terminant sa déclaration par CLOSED:

```
PROC TOTO (        ) CLOSED
```

Enfin il y a la récursivité. Donc on peut dire que, dans le domaine des sous-programmes, on atteint le niveau de PASCAL.

Autres particularités

Les particularités de COMAL seront étudiées en détail dans nos prochains numéros. Ici, nous n'avons que la place de dire que le souci constant des concepteurs de COMAL a été de conserver la simplicité de BASIC en améliorant quelques détails par endroits.

Ainsi la puissance du traitement de chaînes de BASIC est intégralement conservée, mais, cette fois, le nombre maximum de caractères doit être déclaré par DIM

```
DIM A$ OF 15;
```

L'extraction de sous-chaîne est plus simple qu' en BASIC: A$(5:3) s'écrirait en BASIC MID$(A$,5,3)

Les ordres d'écriture et de lecture de fichiers disques existent sous la forme OPEN, READ FILE, WRITE FILE, CLOSE. Ils peuvent manipuler les fichiers relatifs.

D'autres simplifications de détail sont, par exemple l'écriture A:+1 à la place de A:=A+1 ou encore le fait que les instructions peuvent recevoir un label:

```
ICI: A:=A-2*B
```

et les GOTO se réfèrent à ces labels symboliques et non aux numéros de ligne: GOTO ICI. C'est infinîment plus parlant.

L'éditeur COMAL

Comme pour BASIC, les programmes COMAL sont entrés sous le contrôle d'un éditeur qui a certains avantages par rapport à BASIC.

- Il y a une numérotation automatique. (AUTO)

- Quelle que soit la manière dont une instruction a été entrée, on a au listing une indentation qui met la structure en évidence.

- Il y a moyen de sauvegarder sur disque des morceaux

de programme et surtout de charger un programme par morceaux à partir du disque, donc de gérer des bibliothèques de sous-programmes usuels.

- Enfin, l'avantage le plus important, surtout pour les débutants, est que les instructions sont examinées lors de la frappe et que le système n'accepte pas une instruction qui n'est pas syntaxiquement correcte. On n'a pas à attendre l'exécution pour avoir certaines fautes signalées.

Conclusion

Après quelques essais, nous pensons que COMAL fera un "malheur" surtout en enseignement, auprès des débutants. En effet, tout en conservant la simplicité de BASIC, il apporte certains avantages dont les plus essentiels sont:

- programmation structurée

- mécanisme de sous-programmes évolué

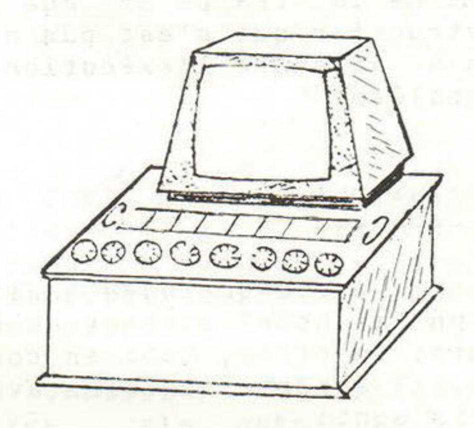
-examen de la correction des instructions lors de la frappe.

Il est bien possible que COMAL supplante BASIC, d'autant que l'adaptation à COMAL de programmes BASIC existants est très simple.

Si vous hésitez entre BASIC et PASCAL, adoptez COMAL.

Daniel-Jean DAVID

```
*******************
*                 *
* LE PASCAL DU CBM *
*                 *
*******************
```



Présentation

PASCAL est un langage évolué dont l'implémentation sur micro-ordinateur est très efficace et qui offre de nombreux avantages par rapport à BASIC.

- -une structure de blocs du type ALGOL
- -une structuration des données extrèmement puissante
- -la récursivité
- -l'allocation dynamique des variables
- -une grande lisibilité des programmes.

Créé par N. WIRTH (Zürich,Suisse) à la suite des études sur la programmation structurée, PASCAL se compose de deux parties:

-un compilateur qui traduit le programme dans un langage machine appelé programme code (p-code)
-un interpréteur qui simule un programme machine et exécute le programme code.

Il existe une autre version, le PASCAL UCSD (Université de Californie à San Diego), dont le p-code est plus facile à interpréter et qui a donné lieu à la réalisation d'un processeur ayant le programme code comme jeu d'instructions (PASCAL Microengine de Western Digital).

Le PASCAL COMMODORE a été écrit en Angleterre par TRANSAM COMPONENTS LIMITED et est du type Zürich.

Fonctionnement

Il nécessite l'utilisation d'un CBM 3032 et d'un disque 3040. Il peut fonctionner sous deux modes:

-un mode résident dans lequel le compilateur et l'interpréteur résident simultanément en mémoire . Ce mode permet d'apprendre le langage et d'écrire de petits programmes n'utilisant pas les disquettes.

-un mode disque dans lequel on appelle le compilateur et l'interpréteur séquentiellement.

Dans les deux modes, PASCAL utilise l'éditeur du CBM avec tous ses avantages et quelques fonctions supplémentaires:

-Auto: numérotation automatique

-Upper, lower: mode majuscule ou minuscule

-Break: appel du moniteur langage machine (.X retourne à PASCAL)

-Number: rénumérotation

-Find: recherche d'une chaîne

-Change: recherche et modification d'une chaîne

-Delete: effacement des lignes (mêmes paramètres que List)

-Hex: conversion décimal-hexadécimal

-Decimal: conversion hexa-décimal

-Basic: retour à BASIC (nécessite ↑PASCAL pour revenir à PASCAL)

-Put, get: pour sauver ou lire un programme sur disquette (le programme PASCAL est sous forme de fichier séquentiel).

En mode résident:

-R: run (compilation et exécution)
-L: compilation et listing sur l'écran
-P: compilation et listing sur l'imprimante
-Disk: passage au mode disque (seul l'éditeur reste en mémoire).

En mode disque:

-Resident: retour au mode résident
-Comp nom de fichier: compilation
-Ex nom de fichier: exécution
-Link: permet de chaîner des programmes (compilation séparée)
-Locate: permet de créer un programme exécutable à partir du BASIC (interpréteur + p.code)

En plus, les commandes BASIC suivantes sont permises en mode direct:

PRINT(ou ?), PRINT#, OPEN, CLOSE, CMD, POKE, SYS, FOR, LET ainsi que les commandes du DOS SUPPORT (>,↑,/).

Avantages et extensions du PASCAL-CBM

Il s'agit d'extensions par rapport au PASCAL standard tel qu'il a été définit par WIRTH,et qui permettent d'utiliser les avantages du CBM.

-Spécification des noms de fichiers

-Lecture de chaîne de caractères

-Utilisation des nombres hexadécimaux;très utile pour les entrées-sorties:

```
const portA=$e84f;
.
.
begin

        poke(portA,$3f);
```

-Accès à la mémoire d'écran:

.La procédure VDU(x,y:integer;c:char) mémorise le caractère c à la ligne x,colonne y de l'écran.

.Page efface l'écran.

.La fonction getkey: char permet d'obtenir le caractère lu directement sur le port du clavier. S'il n'y a pas de caractère on obtient la valeur chr(0); donc pour attendre la frappe d'un caractère on écrira:

```
while getkey =chr(0)do;
```

-Entrées-sorties hexadécimales:

.wrhex (f:text;x:integer) écrit x sous forme de 4 chiffres hexadécimaux sur le fichier f.

.wrhex2 (f:text;x:0..255) écrit l'octet x sous forme hexadécimale sur f.

exemple:

```
rewrite(printer,4,0)
wrhex(printer,-1);wrhex2(output,3)
```

écrira FFFF sur l'imprimante et 03 sur l'écran

.rdhex(f:text):integer ;cette fonction lit une valeur sur 16 bits (4 chiffres hexadécimaux) sur le fichier f

-Fonctions binaires: ANDB, ORB, XORB, NOTB, SHL et SHR:

Ce sont des fonctions logiques qui opèrent sur des entiers traités sur 16 bits.

-Masquage des erreurs d'entrées-sorties:

.iotrap(false) protège le programme d'une sortie innoportune due à une entrée invalide.

.iotrap(true) restaure le fonctionnement normal.

.ioerror: integer donne le type d'erreur.

-Touche STOP:

Breaks(true) et breaks(false) autorise et interdit la touche STOP.

-Générateur de nombre aléatoire:

random: 0..255

-Horloge interne:

trois fonctions sont prédéfinies:

.hours: integer

.minutes: integer

.seconds: integer

exemple:

```
settime(13,30,00);
```

```
writeln(hours,':',minutes,':',seconds);
affichera:   13:  30:  0
```

-Chaînage des programmes:

chain(nom de fichier) arrête l'exécution du programme en cours, charge le programme référencé et le lance

-Interface avec des programmes écrits en langage machine:

Les programmes écrits en langage machine sont déclarés en PASCAL-CBM comme des fonctions ou des procédures dont le corps est remplacé par le mot "extern" suivi de l'adresse en hexadécimal.Les paramètres sont transmis par pile.

-Compilation séparée:

Le PASCAL-CBM permet de compiler un programme en plusieurs morceaux et de le réunir par une seule commande. Ceci est très utile pour les gros programmes.

CONCLUSIONS

Le PASCAL-CBM est très satisfaisant:

-La version du langage est complète et,avec les programmes de gestion de chaînes de caractère qui sont livrés avec la disquette,est supérieure à la version UCSD.

-Les extensions permettent de profiter des ressources du CBM, notamment l'IEEE.

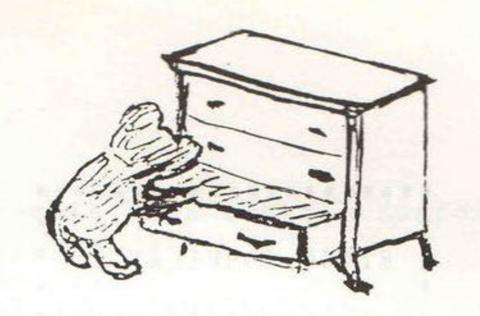
-Les fonctions de manipulation de bits et la possibilité d'écrire des sous-programmes en langage machine en font un véritable PLM.

-Son utilisation s'avère agréable (DOS SUPPORT, répétition sur les touches, fonctions de confort...), mais attention:ne pas oublier de sauver un programme après édition et avant compilation!

-En moyenne les programmes s'exécutent plus vite qu'en BASIC.

Avec de tels avantages, il serait souhaitable de disposer d'un véritable compilateur (traducteur code-p, code 6502) permettant une bonne utilisation en temps réel.

Jean-Luc DESCHAMPS



```
**********************************
*                                *
* COMMENT ACCELERER L'EXECUTION  *
*                                *
*    DE VOS PROGRAMMES BASIC     *
*                                *
**********************************
```

Le BASIC du P.E.T./C.B.M. est un des plus rapides du marché (ce fait est bien connu). Il surclasse nettement ses concurrents directs et il est vrai qu'il est remarquable pour un BASIC interprété.

Un certain nombre de précautions ou d'astuces d'écriture des programmes vont être décrites dans cet article et les suivants de cette série, qui vous permettront d'augmenter encore les performances de vos programmes BASIC, de façon assez sensible.

Toutefois, notez bien que les recommandations qui vont être présentées vont à l'encontre de la structuration du programme et de sa lisibilité. Vous avez donc un choix à faire et il est conseillé aux débutants de les suivre avec prudence.

Pour savoir comment accélérer l'exécution des programmes, il faut comprendre par quels mécanismes cette exécution est ralentie lorsqu'on utilise un interpréteur BASIC au lieu du langage machine. On peut dire que ces mécanismes se décomposent en deux étages:

1) étage compilation qui résulte du fait qu'on utilise un langage évolué au lieu du langage machine. Les antidotes que nous proposerons à ce niveau seront valables aussi pour BASIC compilé.

2) étage interprétation qui vient de ce que c'est un interpréteur qui est utilisé, ce qui occasionne des pertes de temps supplémentaires.

```
**********************
*                    *
* ETAGE COMPILATION *
*                    *
**********************
```

Les pertes de temps qui apparaissent à ce niveau viennent toutes du fait que, pour éviter de devenir d'une complexité inextricable, un compilateur (ou un interpréteur) est obligé d'avoir une attitude stéréotypée standard vis-à-vis de votre programme. Il ne peut pas traiter de façon particulière tous les cas particuliers.

Exemple

Si vous écrivez Y=X↑2 , ceci sera exécuté sous la forme Y=exp(2*log(X)) exactement comme si la puissance avait été fractionnaire. Très peu de compilateurs prennent la peine de traiter de façon particulière la puissance entière et de remplacer votre expression par Y=X*X.

Autre exemple

Si dans une expression arithmétique, vous avez plusieurs fois la même sous-expression, le compilateur ou l'interpréteur ne le reconnaitra pas et donc le calculera à chaque fois.

```
  Ex: Y=(X+1)*(X-1)-K*LOG(X+1)
Facilitez les choses en définissant    U=X+1
                                       Y=U*(X-1)-K*LOG(U)
```

Ceci nous amène à nous rendre compte de ce que le compilateur ou l'interpréteur fait exactement ce qui est écrit, sans optimiser (sauf compilateurs de très grosses machines).

Si vous calculez une constante dans une boucle, celle-ci sera recalculée à chaque itération:

```
FOR I=........
E=EXP(1)
...
NEXT
```

remède: sortez la constante de la boucle!

```
E=EXP(1)
FOR I=.......
...
NEXT
```

Voici un exemple plus subtil. Supposons le test de deux conditions en cascade:

```
IF C1 AND C2 THEN ...
```

Cette écriture implique l'évaluation de C1 et de C2 et l'opération AND tandis que si l'on écrit

```
IF C1 THEN IF C2 THEN ...
```

on passe à la suite dès que C1 est fausse.

Pour aller plus loin, il faut réfléchir avec soin à la structure de son programme. C'est toujours recommandé, et même, avant d'être fier d'optimiser une opération, il faut toujours se demander si elle est bien nécessaire.

Voici un exemple très classique: on recherche si K est présent dans un tableau C(N). La première écriture qui vient à l'esprit est:

```
100 FOR I=1 TO N
110 IF C(I)=K THEN PRINT "OUI":GOTO 140
120 NEXT I
130 PRINT "NON"
140.......suite.......
```

Eh bien cette écriture implique deux tests par itération: le test d'égalité et le test d'achèvement de la boucle.

Si l'on dimensionne C à N+1 et qu'on écrit:

```
100 C(N+1)=K:I=1
110 IF C(I)<>K THEN I=I+1:GOTO 110
120 IF I<N+1 THEN PRINT "OUI":GOTO 140
130 PRINT "NON"
140.......suite.......
```

On n'a plus qu'un test dans la boucle.

```
*************************
*                       *
* ETAGE INTERPRETATION *
*                       *
*************************
```

Une des raisons principales de perte de temps d'un interpréteur sur un compilateur est que l'interpréteur retraduit les instructions à chaque fois qu'il passe dessus. Dans une boucle, les instructions de la boucle sont retraduites à chaque itération.

A cela, nous ne pouvons pas grand-chose, mais il y a d'autres éléments sur lesquels nous pouvons agir. Le point est que à chaque fois qu'il y a référence à une constante, celle-ci est recalculée, à chaque fois qu'il y a référence à une variable, celle-ci est recherchée dans la table des variables et à chaque fois qu'il y a réfé-

rence à une instruction, celle-ci est recherchée parmi la suite des instructions.

Constantes

Le remède pour les constantes est qu'il vaut mieux les éviter. A chaque fois que vous utilisez une constante comme 25497, l'interpréteur va chercher chaque chiffre et effectue le calcul (((2*10+5)*10+4)*10+9)*10+7.

Il est souvent plus rapide d'avoir fait K=25497 et de rechercher K.

Variables

A chaque référence à une variable, celle-ci est recherchée dans une table des variables en mémoire, alors qu'avec un compilateur, on fait référence immédiatement à l'adresse de la variable. Ceci est une des plus grandes sources de perte de temps de BASIC interprété. Dans son livre "La pratique du P.E.T./C.B.M. volume II", D-J DAVID donne l'exemple d'un produit de matrices 41 par 41. Le programme BASIC prend 27 mn contre 2 mn en langage-machine!

La boucle la plus interne de ce produit de matrices est écrite:

```
C(I,J)=0
FOR K=0 TO N
C(I,J)=C(I,J)+A(I,K)*B(K,J)
NEXT K
```

Dans l'instruction de sommation, on a

```
3 recherches de I
2     "       " J
2     "       " K
2     "       " C
1 recherche   " A
1     "       " B
```

au total 11 recherches

Si l'on écrivait

```
Z=0
FOR K=0 TO N
Z=Z+A(I,K)*B(K,J)
NEXT K
C(I,J)=Z
```

on n'aurait plus que 8 recherches. De fait, écrit de cette manière, le programme ne prend plus que 18 mn!

Un autre moyen d'accélérer, lorsqu'on a beaucoup de variables, vient du fait que la recherche est faite

dans l'ordre de la table, c'est-à-dire dans l'ordre chronologique où les variables ont été définies. Lorsqu'on a un certain nombre de variables qui sont plus fréquemment utilisées que d'autres, il est fondamental d'assurer qu'elles soient en tête de la table, en les initialisant au début du programme, par ordre de fréquences d'emploi décroissantes A=0:B=0 ...

Un autre élément est dû au fait que les variables simples et les tableaux sont dans deux zones consécutives en mémoire. Si jamais on crée une variable nouvelle, tous les tableaux déjà crées doivent être déplacés. Par conséquent il est conseillé, lorsqu'on a de grands tableaux, de définir toutes les variables simples en tête du programme. Cela évitera de longs déplacements des tableaux en cours de programme.

Le "garbage collection"

On désigne ainsi la routine qui compacte les chaînes de caractères inutilisées pour récupérer de la place. Elle risque de prendre beaucoup de temps en CBM 3000 (elle a été très améliorée en BASIC version 4.0) lorsque l'on constitue des chaînes de caratères par concaténations successives. On diminue beaucoup ces temps en fractionnant le regroupement ce qui se fait en appelant FRE(0) à intervalles réguliers.

Instructions

Tout GOTO ou GOSUB entraîne une recherche de l'instruction cible en fonction de son numéro. Si le numéro cible est inférieur au numéro de la ligne de départ, on parcourt toute la chaîne des instructions à partir du début. S'il est supérieur, on parcourt la chaîne à partir de la ligne de départ.

On gagne d'autant plus de temps que le parcours est moins long. Cela implique les trois conseils suivants:

1) Avoir le moins de numéros de ligne possibles, donc mettre plusieurs instructions par ligne et éviter les REM. Notez que cela nuit à la lisibilité.

2) Les branchements en avant doivent être courts. Ce n'est pas toujours possible.

3) Les branchements en arrière doivent aboutir près du début du programme: il ne sert à rien qu'une boucle avec GOTO en arrière soit courte, il vaut mieux qu'elle soit

au début du programme.

De même,on a intérêt à mettre au début du programme un sous-programme appelé souvent.

```
  1 GOTO 100                          ...........
 10 .........                         ...........
    .........  est meilleur que     60 GOSUB 1000
100.........                          ...........
    .........                     1000...........
    GOSUB 10                          ...........
```

Tant qu'à faire de mettre des REM, il vaut mieux les mettre en fin de programme, ainsi que les DATA.

```
$$$$$$$$$$$$$$
$            $
$ CONCLUSION $
$            $
$$$$$$$$$$$$$$
```

Tels sont les quelques conseils qu'on peut donner pour exploiter au mieux les particularités de l'interpréteur BASIC. Certains des conseils donnés vont de pair avec un gain de place, comme nous le verrons dans un prochain article; d'autres sont au contraire contradictoires et nécessitent un compromis.

Pour gagner encore plus de temps, un recours existe: BASIC compilé (nous en donnerons un banc d'essai dans le prochain numéro), ou le langage machine.

Enfin d'autres astuces que celles données dans cet article existent certainement. Nous les décrirons dans nos prochains numéros, à mesure qu'elles auront été expérimentées. Nous demandons, bien sûr, à nos Lecteurs, de nous faire part de leurs découvertes, et expériences: elles profiteront certainement aux autres Lecteurs.

Daniel TRECOURT et Pierre Etienne THALBERG

```
*********************************************
*                                           *
* UNE APPLICATION ULTRA-SIMPLE SUR CBM 8000 *
*                                           *
*        UN MINI-ANNUAIRE TELEPHONIQUE      *
*                                           *
*********************************************
```

Voici un programme qui vous permettra de retrouver instantanément les numéros de téléphone de vos amis et aussi de découvrir quelques principes de programmation du CBM.

On suppose que le fichier annuaire existe déjà. Pour le créer avec au départ "zéro" nom, vous tapez en mode direct:

```
DOPEN#2,"FICH-ANNUAIRE,W"
PRINT#2,"0";CHR$(13);:DCLOSE
```

En régime de croisière, votre fichier disque aura la structure suivante: nombre d'individus (N) puis pour chaque individu I son nom (NOM$(I)) et son numéro (X$(I)). N est au maximum 100.

Le programme commence par un appel au sous-programme 1000 qui a pour effet de lire le fichier dans son état actuel en mémoire:

on reconnaît
- l'ouverture du fichier en lecture (1000)
- la lecture du nombre d'individus actuel N
- la boucle qui lit les N couples nom/numéro

Ensuite le programme procède par menu: l'utilisateur devra taper la lettre correspondant à la commande qu'il désire. Les commandes au choix sont:

E entrée de nouveaux noms qui viendront se rajouter à la fin de l'annuaire.

R recherche d'un numéro: vous fournissez un nom et on vous imprime le numéro correspondant, ou le message "pas trouvé" s'il n'est pas dans l'annuaire.

L/P liste de tout l'annuaire sur l'écran (L) ou l'imprimante (P).

S sauvegarde de l'annuaire sur disque (avec les nouveaux noms que l'on a ajoutés).

On trouve de 85 à 130 les branchements vers les différentes commandes.

En 500, l'entrée d'un nouveau nom. N est augmenté de 1.

En 700, listing du fichier. On utilise le fait que l'écran peut se manipuler par PRINT # avec le numéro de périphérique 3. Si la liste est sur écran on fait P=3, si elle est sur imprimante, on fait P=4 puis on parcourt tout le fichier dans une boucle.

Le cas de l'écran pose un problème: il faut laisser le temps de lire les noms d'où l'attente qu'on ait appuyé sur SHIFT (WAIT 152,1) avant de passer à la commande suivante.

EN 800, écriture du fichier sur disque. On crée avec @de remplacement) un fichier de même nom que l'ancien.

On remarque que toutes les impressions se font avec mouvements de curseur qui soignent la mise en page.

Perfectionnement

Le programme tel qu'il est souffre de quelques imperfections (en plus de son caractère simplet):

1) Lorsqu'on fournit une lettre de commande, il faut taper RETURN.

Si l'on remplaçait l'instruction 80 par:

```
80 GET A$: IF A$="" GOTO 80
```

il suffirait de taper la lettre de commande pour déclencher l'exécution.

2) Dans la commande d'entrée des noms, après chaque nom on a le menu et on doit refaire E pour entrer un autre nom.

Au lieu de revenir au menu, on pourrait poser une question du style "tapez A pour un autre ou M pour revenir au menu".

Pour cela, avec l'instruction GET que nous venons de découvrir, nous pouvons écrire:

```
520 PRINT"bbTAPEZ A  POUR UN  AUTRE,M POUR  REVENIR AU
MENU"
530 GET A$:IF A$="" GOTO 530
540 IF A$="A" GOTO 500
550 GOTO 20:REM ON REVIENT AU MENU DES QUE A$<>"A"
```

On pourrait faire la même modification avec la commande R.

3) Le problème du temps de lecture sur écran que nous avons évoqué ne se pose pas qu'à la fin s'il y a plus de 20 noms.

On peut écrire:

```
700 OPEN 1,P      (sans changement)
705 L=0
710 FOR I=1 TO N    (sans changement)
715 L=L+1:IF L=20 AND P=3 THEN GOSUB 750
720 PRINT#1,NOM$(I),X$(I)
730 NEXT I      (sans changement)
740 CLOSE 1:IF P=3 THEN GOSUB 750
745 GOTO 20
750 (sans changement)
760 WAIT 152,1:L=0:RETURN
```

On voit que 750-760 est devenu un sous-programme.

On a introduit un compteur de lignes L. Remarquez la condition composée en 715:

si L=20 (trop de lignes) et P=3 (impression sur écran), alors appeler 750 pour marquer l'attente.

Daniel-Jean DAVID et Jacques COUTHURES

```
10 DIM NOM$(100),X$(100)
15 GOSUB 1000
20 PRINT"{CLR}**************ANNUAIRE ELECTRONIQUE**********"
25 PRINT"TAPEZ.........."
30 PRINT"                    {RVS}E{OFF}    POUR ENTRER DE NOUVEAUX NOMS"
40 PRINT"                    {RVS}R{OFF}    POUR RECHERCHER UN NUMERO"
50 PRINT"                    {RVS}L{OFF}    POUR LISTE SUR ECRAN"
60 PRINT"                    {RVS}P{OFF}    POUR LISTE SUR IMPRIMANTE"
65 PRINT"                    {RVS}S{OFF}    POUR SAUVEGARDE SUR DISQUE"
70 PRINT"{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}{DOWN}"
80 INPUT A$
85 IF A$="S" GOTO 800
90 IF A$="E" GOTO 500
100 IF A$="R" GOTO 600
110 IF A$="L" THEN P=3:GOTO 700
120 IF A$="P" THEN P=4:GOTO 700
130 GOTO 20
500 N=N+1
510 INPUT"{DOWN}{DOWN}NOM,NUMERO";NOM$(N),X$(N)
520 GOTO20
600 INPUT"{DOWN}{DOWN}NOM CHERCHE";C$
610 FOR I=1 TO N
620 IF NOM$(I)=C$ GOTO 650
630 NEXT I
640 PRINT"{DOWN}{DOWN}******{RVS}PAS TROUVE{OFF}******{DOWN}{DOWN}":GOTO20
650 PRINT"{DOWN}{DOWN}"NOM$(I)"          {RVS}"X$(I)"{DOWN}{DOWN}":GOTO20
700 OPEN 1,P
710 FOR I=1 TO N
720 PRINT#1,NOM$(I),X$(I)
730 NEXT I
740 CLOSE 1:IF P=4 GOTO 20
750 PRINT"{DOWN}{DOWN}{DOWN}APPUYEZ SUR SHIFT POUR CONTINUER"
760 WAIT 152,1:GOTO 20
800 DOPEN#2,"@FICH-ANNUAIRE,W":CR$=CHR$(13)
810 PRINT#2,STR$(N);CR$;
820 FOR I=1 TO N
830 PRINT#2,NOM$(I)CR$;:PRINT#2,X$(I)CR$;
840 NEXT:CLOSE2:GOTO 20
1000 DOPEN#2,"FICH-ANNUAIRE"
1010 INPUT#2,N
1020 FOR I=1 TO N
1030 INPUT#2,NOM$(I):INPUT#2,X$(I)
1040 NEXT:CLOSE2:RETURN
READY.
```



```
****************************************
*                                      *
*     PASCAL : UN UTILITAIRE           *
*                                      *
*     POUR LA MISE AU POINT            *
*                                      *
* ET LA DOCUMENTATION DES PROGRAMMES   *
*                                      *
****************************************
```

Ce petit utilitaire a pour but de fournir un listing d'un programme PASCAL en donnant la reférence croisée des noms utilisés dans le programme. La liste des noms de variables et des mots-clefs est classée alphabétiquement et permet l'ajout de nouveaux noms de variables sans interférence avec les précédents.

Ce programme permet de voir la manipulation des structures dynamiques et démontre la "simplicité" d'un classement alphabétique lorsqu'on utilise la récursivité.

On notera la manipulation des entrées-sorties du CBM en PASCAL.

La numérotation des lignes se fait de la même manière que l'éditeur, ainsi il est très facile de retrouver une ligne et une variable. On peut très facilement passer à la numérotation du compilateur, il suffit pour cela de remplacer:

n:=990 par n:=0 aux lignes 1750 et 1860
n:=n+10 par n:=n+1 à la ligne 1870

La gestion des commentaires peut paraître assez compliquée, mais ceci est dû au fait qu'il y a deux caractères pour annoncer un commentaire. Une autre méthode aurait pu consister à lire une ligne en bloc dans un tableau de 80 caractères.

Bon courage et à vos claviers.

Jean-Luc DESCHAMPS

```
1000 program documentation;
1010  (* definition des constantes et des variables *)
1020  const c1=20;
1030        c2=8;
1040        c3=6;
1050        c4=9999;
1060  type alfa=packed array[1..c1] of char;
1070       motref=↑mot;
1080       elementref=↑element;
1090       mot=record cle:alfa;
1100                  premier,dernier:elementref;
1110                  gauche,droite:motref;
1120           end;
1130       element=packed record
1140                  ligne:0..c4;
1150                  prochain:elementref
1160           end;
1170  var racine:motref;
1180      nomfich:packed array[1..80] of char;
1190      ch:char;
1200      k,k1:integer;
1210      n:integer;
1220      id:alfa;
1230      imprimante:text;
1240      f:text;
1250      a:array[1..c1] of char;
1260      sw:integer;
1270  (* debut des procedures *)
1280  (* procedure : imprimer un caractere *)
1290  (*    un changement de code est necesaire pour certains caracteres *)
1300  procedure impri_car(var c:char);
1310   begin
1320   if (c='@') or (c='[') or (c='\') or (c=']') or (c='↑') or (c='_')
1330   then write(imprimante,chr(145),c,chr(17))
1340   else write(imprimante,c)
1350   end;
1360  (* procedure : entree dans la liste d'un nouveau mot *)
1370  procedure recherche(var w1:motref);
1380    var w:motref;x:elementref;
1390   begin w:=w1;
1400      if w=nil then
1410      begin new(w);new(x);
1420            with w↑ do
1430            begin cle:=id;gauche:=nil;droite:=nil;
1440                  premier:=x;dernier:=x
1450            end;
1460            x↑.ligne:=n;x↑.prochain:=nil;w1:=w
1470      end else
1480      if id<w↑.cle then recherche(w↑.gauche) else
1490      if id>w↑.cle then recherche(w↑.droite) else
1500      begin new(x);x↑.ligne:=n;x↑.prochain:=nil;
1510        w↑.dernier↑.prochain:=x;w↑.dernier:=x
1520      end
1530   end;
1540  (* procedure : impression de la liste des mots et des elements *)
1550  procedure impri_arbre(w:motref);
1560     procedure impri_mot(w:mot);
```

```
        var l:integer;x:elementref;
     begin write(imprimante,' ',w.cle);
           x:=w.premier;l:=0;
           repeat if l=c2 then
                  begin writeln(imprimante);
                    l:=0;write(imprimante,' ':c1+1)
                  end;
               l:=l+1;write(imprimante,x↑.ligne:c3);x:=x↑.prochain
           until x=nil;
           writeln(imprimante)
     end;
   begin if w<>nil then
         begin impri_arbre(w↑.gauche);
               impri_mot(w↑);impri_arbre(w↑.droite)
         end
   end;
  (* debut du programme *)
  (* initialisation des variables et des fichiers *)
  begin racine:=nil;n:=990;k1:=c1;
    sw:=0;
    page(output);
    writeln('nom du programme ? ');
    read(nomfich);
    reset(f,nomfich);
    rewrite(imprimante,4,256);
    writeln(imprimante,chr(147),chr(1),'PROGRAMME ',nomfich,chr(129));
    writeln(imprimante);
    read(f,ch);
    while not eof(f) do
      begin if n=c4 then n:=990;
        n:=n+10;write(imprimante,n:c3);
         write(imprimante,' ');
         while not eoln(f) do
          begin
  (* lecture d'un mot et insertion dans la liste *)
            if ch in['a'..'z'] then
            begin k:=0;
               repeat if k<c1 then
                  begin k:=k+1;a[k]:=ch;
                        write(imprimante,ch);
                        if not eoln(f) then read(f,ch) else ch:=' ';
                  end;
               until not(ch in['a'..'z','0'..'9']);
               if k>=k1 then k1:=k else
               repeat a[k1]:=' ';k1:=k1-1
               until k1=k;
               pack(a,1,id);recherche(racine)
            end else
            begin
  (* test des commentaires *)
              if ch='('then
                 begin impri_car(ch);read(f,ch);sw:=1;
                       if ch='*' then
                          repeat repeat impri_car(ch);read(f,ch)
                                 until ch='*';
                                 impri_car(ch);read(f,ch);
                          until ch=')';
                 end;
  (* test des literaux *)
              if ch=''''then begin sw:=0;
```

```
2170                repeat impri_car(ch);read(f,ch)
2180                until ch='''' end;
2190             if sw=0 then begin impri_car(ch);read(f,ch) end else sw:=0
2200          end
2210        end;
2220        writeln(imprimante,ch);if not eof(f) then read(f,ch)
2230      end;
2240      writeln(imprimante);
2250      writeln(imprimante,chr(1),'REFERENCE CROISEE',chr(129));
2260      writeln(imprimante);
2270      impri_arbre(racine);
2280  end.
```

Voici, à titre d'exemple, le début de la table des références croisées obtenue en appliquant ce programme à lui-même.

REFERENCE CROISEE

a	1250	1950	2010	2030				
alfa	1060	1090	1220					
arbre	1550	1690	1700	2270				
array	1060	1180	1250					
begin	1310	1390	1410	1430	1500	1580	1610	1680
	1690	1750	1860	1900	1930	1950	2050	2080
	2160	2190						
c	1300	1320	1320	1320	1320	1320	1320	1330
	1340							
c1	1020	1060	1250	1620	1750	1940		
c2	1030	1600						
c3	1040	1640	1870					
c4	1050	1140	1860					
car	1300	2080	2100	2120	2170	2190		
ch	1190	1840	1920	1950	1960	1970	1970	1990
	2070	2080	2080	2090	2100	2100	2110	2120
	2120	2130	2160	2170	2170	2180	2190	2190
	2220	2220						
char	1060	1180	1190	1250	1300			
chr	1330	1330	1820	1820	1820	2250	2250	
cle	1090	1430	1480	1490	1580			
const	1020							
dernier	1100	1440	1510	1510				
do	1420	1850	1890					
documentation	1000							
droite	1110	1430	1490	1700				
element	1080	1130						
elementref	1080	1100	1150	1380	1570			
else	1340	1470	1480	1490	1970	2000	2040	2190
end	1120	1160	1350	1450	1470	1520	1530	1630
	1670	1710	1720	1980	2040	2140	2180	2190
	2200	2210	2230	2280				
eof	1850	2220						
eoln	1890	1970						
f	1240	1800	1840	1850	1890	1970	1970	2080
	2100	2120	2170	2190	2220	2220		
gauche	1110	1430	1480	1690				
id	1220	1430	1480	1490	2030			
if	1320	1400	1480	1490	1600	1680	1860	1920
	1940	1970	2000	2070	2090	2160	2190	2220
impri	1300	1550	1560	1690	1700	1700	2080	2100
	2120	2170	2190	2270				
imprimante	1230	1330	1340	1580	1610	1620	1640	1660
	1810	1820	1830	1870	1880	1960	2220	2240
	2250	2260						

```
****************************************************
*                                                  *
* UN TRUC SENSATIONNEL: SPOOLING DISQUE-IMPRIMANTE *
*                                                  *
****************************************************
```

Le CBM vous permet de faire une chose qui n'est possible normalement que sur grosses machines. Vous pouvez faire autre chose en même temps que se déroule une longue sortie sur imprimante. Sur grosses machines, toutes les sorties se font sur disque (donc vite), puis une tâche auxiliaire appelée le SPOOL fait le transfert disque-imprimante, en simultanéité avec d'autres traitements.

Grace à l'IEEE et à ses périphériques intelligents, le CBM vous permet de procéder exactement de la même manière.
Supposons que nous ayons un long listing à faire;
on fait:

1)
```
OPEN 8,8,8,"0:LI,S,W"
CMD 8:LIST
PRINT#8:CLOSE 8
```

Ceci effectue le listing sur disque.

2) Charger le programme que l'on voudra exécuter pendant la sortie sur imprimante.

3)
```
OPEN 8,8,8,"0:LI"
POKE 165,72:SYS A:POKE165,104:SYS A
OPEN 4,4:CMD 4:POKE 176,3:POKE 174,0
```

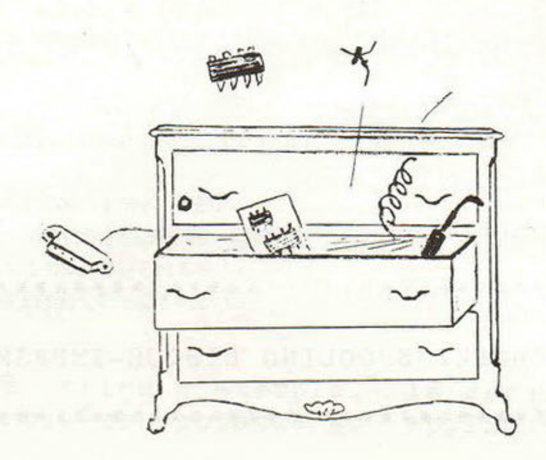
REMARQUE: A vaut 61668 sur CBM 3000 et 61695 sur CBM 4000/8000.

4) Le listing démarre. Faire fonctionner le programme chargé en 2. Attention: celui-ci ne doit en aucune façon utiliser le bus IEEE.

5) Lorsque tout est terminé:

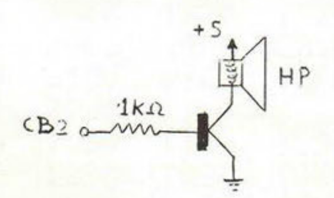
```
PRINT#4:CLOSE 4:OPEN 8,8,8:CLOSE 8
```

Pierre-Etienne THALBERG



```
****************************************
*                                      *
* L'INTERFACE SONORE SUR LE PET/CBM *
*                                      *
****************************************
```

Beaucoup de montages sont possibles pour commander un haut-parleur à partir de votre PET/CBM. Le montage le plus répandu correspond au schéma ci-dessous:



Les créneaux sont captés sur la broche CB2 (borne M du connecteur du port utilisateur) du boîtier 6522. La masse est prise à la borne N du même connecteur. Le signal amplifié par le transistor 2N2222 (ou n'importe quel type npn) passe au +5V à travers la bobine mobile du haut-parleur. Le 5V est pris sur le connecteur du magnétophone. Il constitue une alimentation parfaite tant qu'on ne demande pas trop de courant. Notons que sur CBM, on peut en obtenir plus que sur les anciens P.E.T.

On peut insérer un potentiomètre pour avoir un réglage de l'intensité sonore. Un petit condensateur (0,1 µF) en parallèle sur le 1k rend le son plus sinusoidal.

Programmation

Ce branchement est le plus employé car il est le plus facile à programmer: il suffit de quelques POKE. La

génération des créneaux est automatique alors que dans d'autres montages, il faut générer complètement les créneaux et les délais.

En effet le boîtier 6522 contient un registre à décalage de 8 bits à l'adresse 59466 ($E84A). Si l'on met dans ce registre un motif binaire M formé de 0 et de 1, on aura des créneaux à la mesure des décalages. Les décalages sont activés si l'on met (par POKE) 16 dans le registre de commande du 6522, d'adresse 59467 ($E84B). les décalages sont inhibés, donc le son est arrêté si l'on met 0 dans ce registre.

La fréquence de décalage est déterminée par la valeur T qu'on met en 59464 ($E848): ce registre est le registre de comptage d'un des temporisateurs du 6522. Une fois les décalages lancés, le phénomène continue de lui-même: il y a en fait rotation dans le registre à décalage de sorte que lorsque 8 bits ont été décalés, le même motif binaire M est prêt à être décalé de nouveau.

La fréquence du son est déterminée à la fois par M et T. On comprend bien que,à T égal,si M=00110011 (51), la fréquence sera double de ce qu'on a pour M=00001111 (15). Elle sera encore doublée pour M=01010101 (85). C'est un moyen simple pour faire des octaves.

Le motif M détermine aussi le timbre: M=15 donne un créneau simple. M=00111111 (63) introduit de l'harmonique 2 (cela ne transforme tout de même pas votre CBM en violon!).

La gamme

En résumé, pour obtenir un son, il suffit d'écrire:

POKE 59466,M : POKE 59464,T : POKE 59467,16

et pour l'arrêter:

POKE 59467,0 (sinon il continue indéfiniment).

Voici, pour M = 15, les valeurs de T qui donnent la gamme

ut-3	242	sol	162
ut#	228	sol#	152
ré	215	la 440Hz	144
ré#	203	la#	136
mi	192	si	128
fa	181	do	121
fa#	171		

Pierre-Etienne THALBERG



UN PIEGE DE CMD UTILISE EN CONJONCTION AVEC LE MONITEUR LANGAGE MACHINE

Sur CBM 3000, lorsque vous vouliez obtenir les sorties sur imprimante, vous faisiez le classique:

OPEN 4,4 : CMD 4

puis vous lanciez le moniteur langage machine par SYS 4 (où on trouve un BREAK).

Sur CBM 4000 et 8000, cela ne marche pas. En effet un BREAK annule l'effet de CMD dans la version BASIC 4.0. Il faut entrer dans le moniteur langage machine, non par un BREAK, mais par l'entrée normale, donc par SYS 54386.

ADRESSES ABSOLUMENT FONDAMENTALES

adresse	ancien P.E.T.	PET/CBM 3000	CBM 4000/8000	VIC
CHRGET	\$C2 194	\$70 112	\$70 112	\$73 115
CHRGOT	\$C8 200	\$76 118	\$76 118	\$79 121
moniteur lang. mach.	--	\$FD11 64785	\$D472 54386	--
RESET tiède impri.READY	\$C38B 50059	\$C389 50057	\$B3FF 46079	\$C474 50292
RESET	\$FD38 64824	\$FCD1 64721	\$FD16 64790	?



```
************************************************
*                                              *
* LES JEUX COUSUS-MAIN : LE JEU DU PENDU       *
*                                              *
************************************************
```

Dans chaque parution de votre nouvelle revue vous trouverez une rubrique destinée à complèter vos connaissances, mais aussi à vous distraire.

Pour le premier numéro, nous vous proposons le jeu du Pendu dont tous les écoliers se souviennent. A l'intention des élèves studieux qui n'y ont jamais joué pendant un cours de géographie ou de mathématiques, en voici les règles:

Vous devez deviner un mot dont vous ne connaissez au début que le nombre de lettres qui le constituent. Si vous trouvez l'une des lettres du mot, celle-ci est placée au bon endroit dans le mot. Si vous choisissez une lettre inutilisée le dessin d'un pendu commence à se former sur l'écran de l'ordinateur; vous avez droit seulement à 4 erreurs, à la cinquième vous avez perdu ... et le mot mystérieux s'affiche au pied de la potence.

Nous vous donnons ci-après le "listing" de ce programme qui fonctionne sans problème sur P.E.T 2001 et sur CBM 3008 - 3016 et 3032. Les touches "curseur" sont indiquées en clair et entre crochets. Le "crochet ouvert" est précédé d'un espace et le "crochet fermé" est suivi également d'un espace dont il n'y a pas lieu de tenir compte; tous les autres espaces des instructions PRINT doivent être scrupuleusement tapés au clavier.

Les mots à trouver sont en DATA (lignes 5000 à 5110). Ils sont choisis de façon aléatoire (ligne 110); bien entendu il vous est possible de les changer en modifiant les lignes de DATA. Nous avons prévu un choix de 50 mots mais vous pouvez en ajouter dans de nouvelles lignes de DATA à condition de remplacer (ligne 110) le nombre 50 par le nouveau nombre de mots.

Amusez-vous bien ...

Jacques COUTHURES

JEU DU PENDU


```
10  DIM L$(20): DIM T(50)
20 J = 1
60  PRINT " [CLR] "; TAB( 13);" [RVS] JEU [OFF]    [RVS] DU [OFF]
    [RVS] PENDU [OFF] "
70 C$ = ""
74 P$ = C$
80  FOR I = 1 TO 20
85 L$(I) = "-"
90  NEXT I
100 K = 0:X = 0
110 A =  INT (50 * ( RND (1))) + 1
120  FOR I = 1 TO J
130  IF T(I) = A THEN 110
140  NEXT I
150 T(I) = A
160  FOR I = 1 TO A
170  READ A$
180  NEXT I
190  RESTORE
200 L =  LEN (A$)
210  PRINT " [DOWN]  [DOWN]  [DOWN] "; TAB( 5);"▀▀▀▀▀▀▀▀▀▀"
220  PRINT  TAB( 5);" [RVS]   [OFF]      [RVS] ◤ [OFF] ◤"
230  PRINT  TAB( 5);" [RVS]   [OFF]     [RVS] ◤ [OFF] ◤"
240  PRINT  TAB( 5);" [RVS]   [OFF]    [RVS] ◤ [OFF] ◤"
250  PRINT  TAB( 5);" [RVS] ◤ [OFF] ◤"
260  PRINT  TAB( 5);" [RVS]   [OFF] ◤"
270  FOR I = 1 TO 8
280  PRINT  TAB( 5);" [RVS]   [OFF] "
290  NEXT I
300  PRINT "      [RVS] ◤ ◥"
310  PRINT "     [RVS] ◤ [OFF] ◤ [RVS]    [OFF] ◥ [RVS] ◥"
320  PRINT "    [RVS] ◤ [OFF] ◤  [RVS]    [OFF]  ◥ [RVS] ◥"
330  PRINT "   [RVS] ◤ [OFF] ◤   [RVS]    [OFF]   ◥ [RVS] ◥"
340  PRINT " [RVS]              "
360  PRINT " [HOME] ";" [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN] "
370  PRINT  TAB( 20);"MOT DE ";L;"LETTRES"
380  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN] "
400  PRINT  TAB( 20);"VOUS PROPOSEZ ";
410  INPUT R$
420 K = K + 1:A = 0
430 C$ = C$ + R$
450  FOR I = 1 TO L
460 B$ =  MID$ (A$,I,1)
470  IF B$ <  > R$ THEN 510
480 A = 1
490 L$(I) = R$
500  GOTO 520
510  GOTO 520
520  NEXT I
530  IF A = 1 THEN 1000
540 X = X + 1
550  PRINT " [HOME] "
560  IF X > 1 THEN 640
570  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN] ";
590  PRINT  TAB( 13);"|"
```

```
600  PRINT  TAB( 12);" ┴ "
610  PRINT  TAB( 12);"|┴|"
620  PRINT  TAB( 12);" ┬ "
630  GOTO 1000
640  IF X > 2 THEN 720
650  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN] ";
660  PRINT  TAB( 11);"/‾‾‾"
680  PRINT  TAB( 10);"//"
690  PRINT  TAB( 9);" || "
700  PRINT  TAB( 9)" └┘ "
710  GOTO 1000
720  IF X > 3 THEN 790
730  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN]  [DOWN] ";
740  PRINT  TAB( 12);"| . |"
750  PRINT  TAB( 12);"| . |"
760  PRINT  TAB( 12);"| . |"
770  PRINT  TAB( 12);"\_/"
780  GOTO 1000
790  IF X > 4 THEN 860
800  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN] ";
810  PRINT  TAB( 15);"\"
820  PRINT  TAB( 15);"\\"
830  PRINT  TAB( 15);" || "
840  PRINT  TAB( 15);" └┘ "
850  GOTO 1000
860  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN] ";
870  PRINT TAB( 12);"/_\"
880  PRINT  TAB( 11);" |||"
890  PRINT  TAB( 11);" |||"
900  PRINT  TAB( 11);" └┘└┘"
910  PRINT " [DOWN]  [DOWN]  [DOWN] "; TAB( 20);"DE PROFONDIS !!!!!"
920  GOTO 2000
1000  FOR I = 1 TO L:P$ = P$ + L$(I): NEXT I
1001  IF A$ = P$ THEN 1500
1002 P$ = ""
1005  PRINT " [HOME] ";
1010  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN] "
1020  PRINT  TAB( 20);"LETTRES UTILISEES"
1025  PRINT  TAB( 19);
1030  FOR I = 1 TO 10
1040  PRINT " "; MID$ (C$,I,1);
1050  NEXT I
1055  PRINT
1060  PRINT  TAB( 20);
1070  FOR I = 11 TO 19
1080  PRINT " "; MID$ (C$,I,1);
1090  NEXT I
1100  PRINT : PRINT
1110  PRINT : PRINT  TAB( 20);
1120  FOR I = 1 TO L
1130  PRINT L$(I);
```

PENDU <SUITE>


```
1140  NEXT I
1150  PRINT : PRINT
1160  GOTO 400
1500  PRINT " [HOME] ";: PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN]  [DOWN]  [DOWN] [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]
     [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN] ";
1510  PRINT  TAB( 10);A$;
1520  PRINT  TAB( 20);"  VOUS GAGNEZ..!"
1530  GOTO 2000
2000  PRINT
2005  PRINT " [UP] "; CHR$ (18);A$
2010  PRINT  TAB( 16);"AUTRE ESSAI (O,N) ";
2020  INPUT T$
2030  IF T$ = "N" THEN 3000
2040 J = J + 1
2050  IF J > 49 THEN 4000
2060  GOTO 60
3000  PRINT " [CLR] "
3010  PRINT " [DOWN]  [DOWN]  [DOWN]  [DOWN]  [DOWN] ";
3020  GOTO 4020
4000  PRINT " [CLR] "
4010  PRINT " [DOWN]  [DOWN]  [DOWN] ";"      LISTE EPUISEE, .........."
4020  PRINT : PRINT "         A TRES BIENTOT"
4030  END
5000  DATA FORT,KAOLIN,WIGWAM,PLASTIQUE,CINEMA
5010  DATA JARS,MAISON,JARDIN,HEXAGONE,PRIMAIRE
5020  DATA ZOOSPORE,ZEROTAGE,YPREAU,XERANTHEME,
5030  DATA WASSINGUE,VULCANISATION,SYNTHETIQUE
5040  DATA SYBARITE,ROENTGENTHERAPIE,RIDEAU,RIDOIR
5050  DATA RAJA,RAMASSIS,OIGNON,OLEICOLE,OZOKERITE,OXALATE,JASPURE
5060  DATA JAQUEMART,JOCRISSE,JOUVENCELLE,HAUSSIER
5070  DATA HAUTURIER,HEMEROTHEQUE,CONJONCTION,COURGE,COUTELAS,CREMONE
5080  DATA ARPETE,ARQUEBUSE,AMORPHE,ANASTIGMATE,AFFINEUR,BONJOUR
5090  DATA AMOUR,BAVETTE,CLAMEUR,ECHAFAUD,ECHAUGUETTE,ECHINODERME
5100  DATA MISANTHROPIQUE,MONITEUR,BUONIVOQUE,MUCILAGE
5110  DATA MUSCARDIN,OUTRECUIDANCE,PARCOMETRE,PARISYLLABE,PAREGORIQUE
```

```
******************************
*                            *
* S'ABONNER GRATUITEMENT ? *
*                            *
******************************
```

Comment vous abonner gratuitement pour votre 2 ème année ? C'est très simple ! Il suffit de nous renvoyer cette feuille complètement remplie accompagnée du réglement, avec votre nom et les noms de 5 de vos amis: vos amis recevront leur abonnement pour cette année, et, pour vous, votre abonnement sera automatiquement renouvelé l'année prochaine.

PARRAIN
------- Je désire renouveler mon abonnement à La Commode, en parrainant les cinq amis dont les noms suivent.

nom --

adresse --

----

votre numéro d'abonnement actuel (obligatoire)
----

date ------------- signature ------------

1° PARRAINE
------------ Je désire m'abonner à La Commode (1 an, 4 numéros = 120 F)

nom --

adresse --

----

date ------------- signature ------------

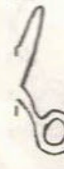
2° PARRAINE
------------ Je désire m'abonner à La Commode (1 an, 4 numéros = 120 F)

nom --

adresse --

----

date ------------- signature ------------



3° PARRAINE
------------- Je désire m'abonner à La Commode (1 an, 4 numéros = 120 F)

nom ---

adresse ---

----

date ------------- signature ------------

4° PARRAINE
------------- Je désire m'abonner à La Commode (1 an, 4 numéros = 120 F)

nom ---

adresse ---

----

date ------------- signature ------------

5° PARRAINE
------------- Je désire m'abonner à La Commode (1 an, 4 numéros = 120 F)

nom ---

adresse ---

----

date ------------- signature ------------

---

ci-joint réglement 5x120 F = 600 F

par CCP CB

N.B. Les cinq abonnements parrainés peuvent être servis à un même établissement d'enseignement. Joindre un bon de commande de l'établissement, et déduire du réglement une remise supplémentaire EDUCATION de 50 F d'où une somme de 550 F.

A envoyer à La Commode
28, rue Vicq d'Azir
75010 PARIS



ADRESSES DES ROUTINES IEEE

2001	3000	4000 8000	ROLE
dec hexa	dec hexa	dec hexa	
61622 FOB6	61622 FOB6	61650 FOD2	Envoi d'une adresse parleur sur le bus IEEE
61626 FOBA	61626 FOBA	61621 FOB5	Envoi d'une adresse écouteur sur le bus IEEE
61740 F12C	61736 F128	61767 F148	Envoi d'une adresse secondaire sur le bus IEEE
61799 F167	61807 F16F	61854 F19E	Envoi d'un caractère sur le bus IEEE
61818 F17A	61823 F17F	61878 F1B6	Envoi de la commande "cessation d'émission" (untalk)
61822 F17E	61827 F183	61881 F1B9	Envoi de la commande "cessation d'écoute" (unlisten)
61831 F187	61836 F18C	61888 F1CO	Acquisition d'un caractère sur le bus IEEE

Pierre-Etienne THALBERG



BIBLIOGRAPHIE

Nos confrères l'Ordinateur Individuel et Microsystèmes ont publié de nombreux articles relatifs au PET/CBM. Voici un rappel de ces articles.

L'Ordinateur Individuel

Le PET au banc d'essai	nº 1	p. 30	oct. 78
Programme de tri par insertion	nº 8	p. 34	juin 79
Application tarification d'assurance	nº 10	p. 52	sept.79
Pendule électronique de table	nº 12	p. 44	nov. 79
Interview de Chuck Peddle	nº 12	p.102	nov. 79
Jeu du Quotient Intellectuel	nº 16	p. 83	avr. 80
Le CBM dans une parfumerie	nº 18	p. 61	juin 80
Syndicat intercommunal des eaux	nº 22	p. 52	nov. 80
Langage machine et assembleur	nº 24	p. 93	fév. 81
	et nº 25	p.109	mars 81

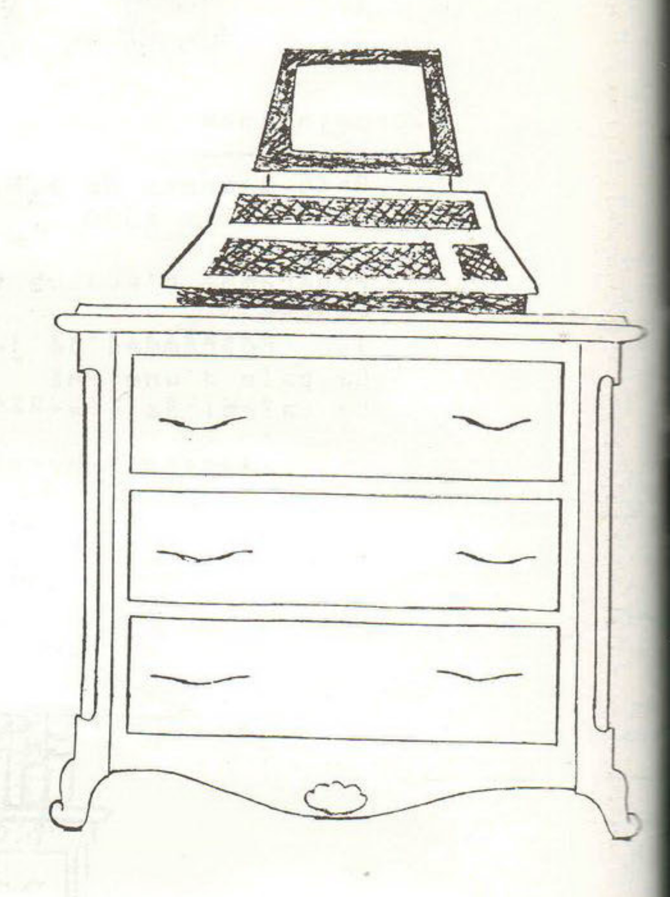
Rappelons qu'à partir du numéro 10 (sept. 79) il y a dans chaque numéro un "A.B.C. du P.E.T.".

Microsystèmes

Banc d'essai du P.E.T.	nº 2	p. 78	nov-déc	78
La famille 6500	nº 8	p. 65	nov-déc	79
et	nº 9	p. 75	jan-fév	80
Programme d'évaluation des ventes	nº 9	p. 30	jan-fév	80
140 programmes de jeux	nº 12	p. 65	juil-août	80
La paie d'une PME	nº 12	p. 89	juil-août	80
Le calcul de l'emprunt	nº 16	p. 62	mars-avr.	81



La Commode



REDACTION

Rédacteur en chef:
Daniel TRECOURT
Chefs de rubriques:
Jacques COUTHURES
Jean DELAVILLE
Jean-Luc DESCHAMPS
Pierre-Etienne THALBERG
Secrétaire de rédaction:
Xavier-Frédéric ARDOUIN
Directeur de la publication:
Daniel-Jean DAVID
Rédaction-vente-abonnements:
28 rue VICQ-d'AZIR
75010 PARIS
Diffusion (boutiques et librairies):
Ed. du P.S.I.
41-51 rue Jacquard
BP86
77400 LAGNY
Publicité:
FORCE 7
41, rue de la Grange aux Belles
75483 PARIS CEDEX 10
Tél 238 66 10
Publié par SEDERMI SARL
28 rue VICQ-d'AZIR
75010 PARIS
Tél 205 87 75

BULLETIN D'ABONNEMENT

à envoyer à La Commode
28, rue VICQ d'AZIR
75010 PARIS

Je désire m'abonner à La Commode (1 an, 4 numéros) à partir du numéro ----------



nom ------------------------------
adresse ------------------------------

----
signature -------------

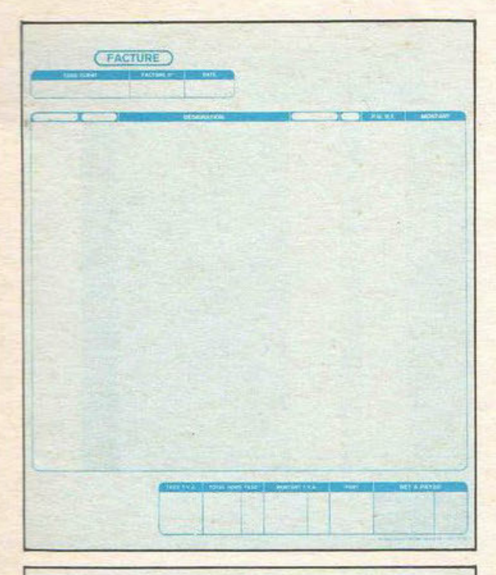
réglement 120 F par

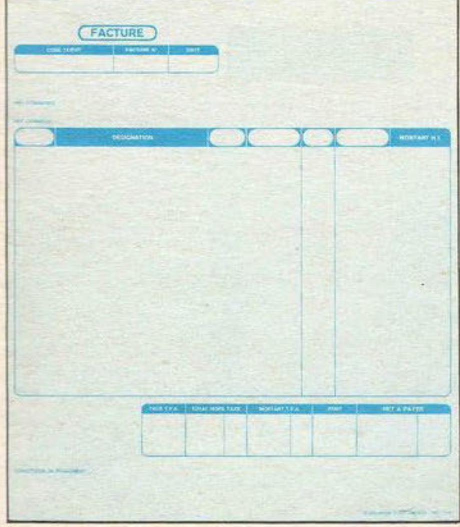
CCP CB ESPECES

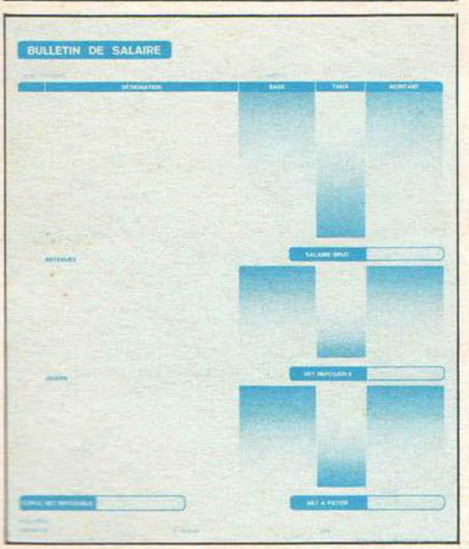
Imprimerie Laballery, 58500 Clamecy. Imprimé en France. Directeur de la publication Daniel-Jean David
Dépôt légal 3e trimestre 1981. Numéro de Commission paritaire : en cours

122, Bd Raspail
75006 PARIS
Tél. : 548.26.24

S.C.A.







Société
Centrale
Achat
Micro
Informatique

VOTRE FOURNISSEUR :

Papier, Disquettes, Librairie, Micros, Cours, que vous soyez Installateur, Revendeur ou bien Utilisateur.

L'EXEMPLE DU MOIS

TARIF UTILISATEUR

	Le MILLE H.T.	
FACTURE 37 LIGNES 240 × 11	**1.098,00**	REF. F 24 3 EXEMPLAIRES
FACTURE 26 LIGNES 240 × 11	**1.098,00**	REF. F 10 3 EXEMPLAIRES
BULLETIN DE PAYE 240 × 11	**695,00**	REF. P 50 2 EXEMPLAIRES

REVENDEURS

CONSULTEZ-NOUS

Coupon à retourner à **SCAMI 122 Bd Raspail - 75006 PARIS**

Société ....

Nom du Responsable .... Tél. ....

Adresse .... Ville ....

Je désire une information sur : ....

....

....